

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: 2003006979 A

(43) Date of publication of application: 10.01.03

(51) Int. Cl

G11B 20/10
G11B 27/00
G11B 27/10
H04J 3/00
H04N 5/765
H04N 5/85
H04N 5/91
H04N 5/92

(21) Application number: 2001189744

(71) Applicant: SONY CORP

(22) Date of filing: 22.06.01

(72) Inventor: KATO MOTOKI

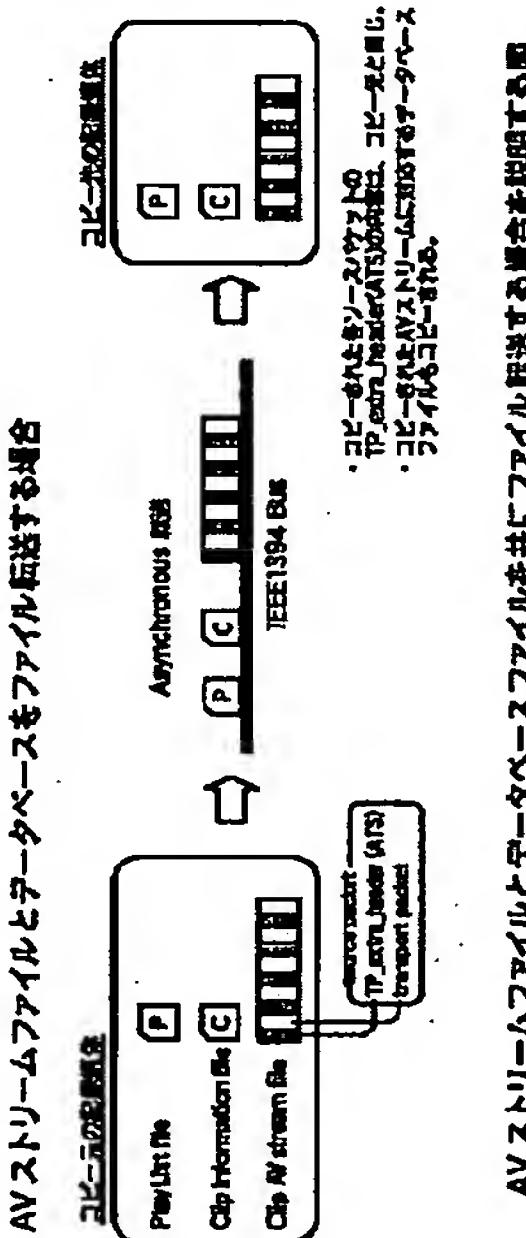
(54) DATA TRANSMISSION EQUIPMENT AND
METHOD, DATA PROCESSOR AND DATA
PROCESSING METHOD, RECORDING MEDIUM,
AND PROGRAM

(57) Abstract:

PROBLEM TO BE SOLVED: To adequately manage the contents of data recorded on a recording medium and reproduction information.

SOLUTION: Clip (Clip Information file and ClipAV stream file) and PlayList are recorded on the recording medium of a copy source. The files of PlayList and Clip are transferred from the copy source through an IEEE 1394 digital bus to the recording medium of a copy destination and copied. The arrival time stamp of the TP_extra_header of each source packet of the copied Clip AV stream file is the same as that of the copy source, and the Clip Information file and the PlayList file corresponding to the copied Clip AV stream file are also copied to the copy destination.

COPYRIGHT: (C)2003,JPO



AVストリームファイルとデータベースファイルを共にファイル操作する場合を示す図

図61

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2003-6979

(P2003-6979A)

(43)公開日 平成15年1月10日 (2003.1.10)

(51)Int.Cl.⁷
G 11 B 20/10
27/00
27/10
H 04 J 3/00

識別記号
3 2 1

F I
G 11 B 20/10
27/00
27/10
H 04 J 3/00

テーマコード^{*}(参考)
D 5 C 0 5 2
3 2 1 Z 5 C 0 5 3
D 5 D 0 4 4
A 5 D 0 7 7
M 5 D 1 1 0

審査請求 未請求 請求項の数27 OL (全 55 頁) 最終頁に続く

(21)出願番号 特願2001-189744(P2001-189744)

(22)出願日 平成13年6月22日 (2001.6.22)

(71)出願人 000002185

ソニー株式会社
東京都品川区北品川6丁目7番35号

(72)発明者 加藤 元樹
東京都品川区北品川6丁目7番35号 ソニ
一株式会社内

(74)代理人 100082131
弁理士 稲本 義雄

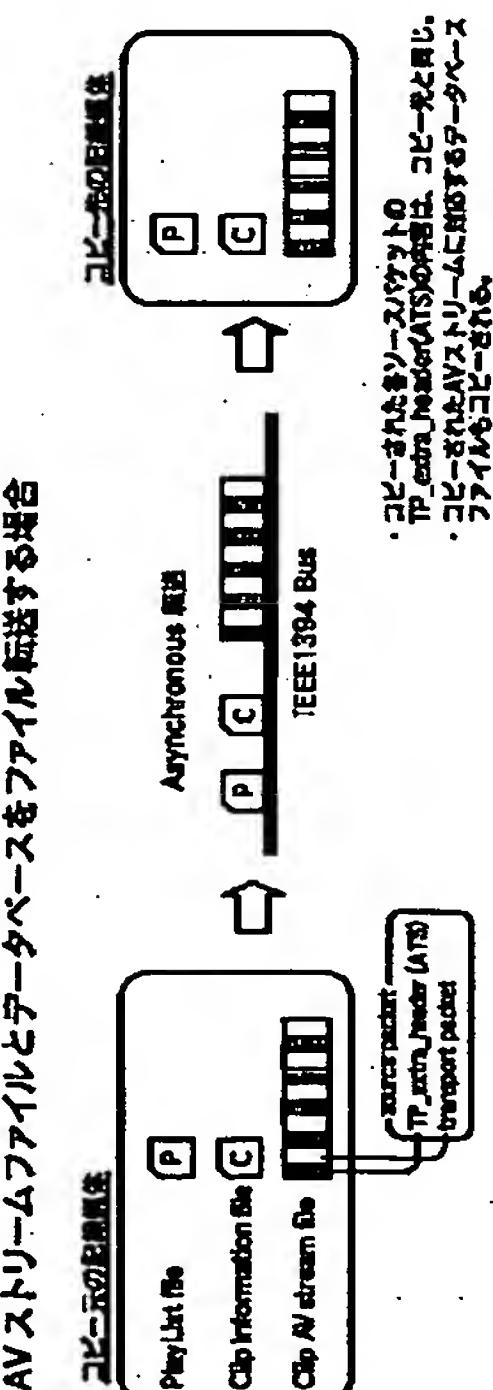
最終頁に続く

(54)【発明の名称】 データ伝送装置および方法、データ処理装置および方法、記録媒体、並びにプログラム

(57)【要約】

【課題】 記録媒体に記録されているデータの内容およ
び再生情報を適切に管理できるようにする。

【解決手段】 コピー元の記録媒体には、Clip (Clip I
nformation fileとClipAV stream file) とPlayListが
記録されている。PlayListとClipのファイルが、コピー
元からコピー先の記録媒体へIEEE 1394のデジタルバ
スを経由して、転送され、コピーされる。コピーされた
Clip AV stream fileの各ソースパケットのTP_extra_he
aderのアライバルタイムスタンプは、コピー元と同じで
あり、また、コピーされたClip AV stream fileに対応
するClip Information fileとPlay Listファイルもコピ
ー先へコピーされる。



AVストリームファイルヒューターベースファイルを共にファイル転送する場合を説明する図

図61

【特許請求の範囲】

【請求項1】 データストリームとその管理情報を記録媒体から読み出す読み出し部と、前記データストリームの中で、指定された再生区間に再生に必要な部分データストリームに対応する部分管理情報を決定する制御部と、前記部分データストリーム、および、前記制御部により決定された部分管理情報を伝送する伝送部とを備えることを特徴とするデータ伝送装置。

【請求項2】 前記伝送部は、さらに前記部分管理情報を、非同期伝送することを特徴とする請求項1に記載のデータ伝送装置。

【請求項3】 前記制御部は、接続された機器の種別に応じて、前記部分データストリームを非同期伝送するか、または同期伝送するかを選択することを特徴とする請求項1に記載のデータ伝送装置。

【請求項4】 前記データストリームは、AVストリームであり、

前記管理情報は、

前記AVストリーム中の符号化情報の不連続点のアドレス情報、

前記AVストリーム中の時刻情報とアドレス情報を関連づける情報、および前記AVストリーム中の特徴的な画像の時刻情報を含むことを特徴とする請求項1に記載のデータ伝送装置。

【請求項5】 前記管理情報は、Clip Informationであり、

前記不連続点のアドレス情報は、SequenceInfoおよびProgramInfoであり、

前記時刻情報とアドレス情報を関連づける情報は、CPIであり、

前記特徴的な画像の時刻情報は、ClipMarkであることを特徴とする請求項4に記載のデータ伝送装置。

【請求項6】 前記部分管理情報は、前記AVストリームの再生区間の指示情報である、PlayListをさらに含むことを特徴とする請求項5に記載のデータ伝送装置。

【請求項7】 前記伝送部は、前記部分データストリームとしての前記AVデータストリームの再生区間の指示情報をさらに伝送することを特徴とする請求項4に記載のデータ伝送装置。

【請求項8】 前記伝送部は、前記AVストリームの再生区間の指示情報の内容を変更しないで、前記部分データストリームとしての前記AVデータストリームの再生区間の指示情報として伝送することを特徴とする請求項7に記載のデータ伝送装置。

【請求項9】 前記伝送部は、前記AVストリームの再生区間の指示情報に関連付けられたサムネール画像、および、前記部分管理情報データに含まれるAVストリーム中の特徴的な画像の時刻情報に関連付けられたサムネール画像をさらに伝送することを特徴とする請求項4に記

載のデータ伝送装置。

【請求項10】 前記AVストリームは、トランSPORTパケットとそのアライバルタイムスタンプから構成されるソースパケットを単位とするデータ列であり、前記部分データストリームとしてのAVストリームは、前記AVストリームのソースパケットのデータ列の部分である、

ことを特徴とする請求項4に記載のデータ伝送装置。

【請求項11】 前記AVストリームは、トランSPORTパケットとそのアライバルタイムスタンプから構成されるソースパケットを単位とするデータ列であり、前記部分データストリームとしてのAVストリームは、トランSPORTパケットを単位とするトランSPORTストリームであることを特徴とする請求項4に記載のデータ伝送装置。

【請求項12】 制御部は、前記AVストリームとその管理情報を共にデータ伝送するか、または前記AVストリームだけをリアルタイム転送するかを切り替えることを特徴とする請求項4に記載のデータ伝送装置。

【請求項13】 データストリームとその管理情報を記録媒体から読み出す読み出しステップと、前記データストリームの中で、指定された再生区間に再生に必要な部分データストリームに対応する部分管理情報を決定する決定ステップと、

前記部分データストリーム、および、前記決定ステップの処理により決定された管理情報を伝送する伝送ステップとを含むことを特徴とするデータ伝送方法。

【請求項14】 データストリームとその管理情報を記録媒体から読み出す読み出しステップと、

前記データストリームの中で、指定された再生区間に再生に必要な部分データストリームに対応する部分管理情報を決定する決定ステップと、

前記部分データストリーム、および、前記決定ステップの処理により決定された管理情報を伝送する伝送ステップとを含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

【請求項15】 データストリームとその管理情報を記録媒体から読み出す読み出しステップと、

前記データストリームの中で、指定された再生区間に再生に必要な部分データストリームに対応する部分管理情報を決定する決定ステップと、

前記部分データストリーム、および、前記決定ステップの処理により決定された管理情報を伝送する伝送ステップとをコンピュータに実行させるプログラム。

【請求項16】 AVストリームを伝送するデータ伝送装置において、

前記AVストリームの伝送先の装置と相互認証する認証部と、

前記認証部による前記相互認証の結果から、前記伝送先の装置が所定のフォーマットに準拠していることが判つ

た場合、前記AVストリームとその管理情報を共にデータ伝送するとともに、前記認証部による前記相互認証の結果から、前記伝送先の装置が前記所定のフォーマットに準拠していないことが判った場合、前記AVストリームだけをリアルタイム伝送する伝送部とを備えることを特徴とするデータ伝送装置。

【請求項17】 AVストリームを伝送するデータ伝送装置のデータ伝送方法において、

前記AVストリームの伝送先の装置と相互認証する相互認証ステップと、

前記認証ステップの処理による前記相互認証の結果から、前記伝送先の装置が所定のフォーマットに準拠していることが判った場合、前記AVストリームとその管理情報を共にデータ伝送するとともに、前記認証ステップの処理による前記相互認証の結果から、前記伝送先の装置が前記所定のフォーマットに準拠していないことが判った場合、前記AVストリームだけをリアルタイム伝送する伝送ステップとを含むことを特徴とするデータ伝送方法。

【請求項18】 AVストリームを伝送するデータ伝送装置のプログラムであって、

前記AVストリームの伝送先の装置と相互認証する相互認証ステップと、

前記認証ステップの処理による前記相互認証の結果から、前記伝送先の装置が所定のフォーマットに準拠していることが判った場合、前記AVストリームとその管理情報を共にデータ伝送するとともに、前記認証ステップの処理による前記相互認証の結果から、前記伝送先の装置が前記所定のフォーマットに準拠していないことが判った場合、前記AVストリームだけをリアルタイム伝送する伝送ステップとを含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

【請求項19】 AVストリームを伝送するデータ伝送装置を制御するコンピュータに、

前記AVストリームの伝送先の装置と相互認証する相互認証ステップと、

前記認証ステップの処理による前記相互認証の結果から、前記伝送先の装置が所定のフォーマットに準拠していることが判った場合、前記AVストリームとその管理情報を共にデータ伝送するとともに、前記認証ステップの処理による前記相互認証の結果から、前記伝送先の装置が前記所定のフォーマットに準拠していないことが判った場合、前記AVストリームだけをリアルタイム伝送する伝送ステップとを実行させるプログラム。

【請求項20】 AVストリームファイル、および前記AVストリームの再生方法を指定するPlayListファイルが記録されている記録媒体から、データを受信するデータ処理装置において、

前記PlayListファイルを受信する受信部と、

前記PlayListを記録するとともに、記録されているPlayListを管理する管理情報ファイルに、前記PlayListファイルに関する情報を追加する記録部とを備えることを特徴とするデータ処理装置。

【請求項21】 AVストリームファイル、および前記AVストリームの再生方法を指定するPlayListファイルが記録されている記録媒体から、データを受信するデータ処理装置のデータ処理方法において、

前記PlayListファイルを受信する受信ステップと、

10 前記PlayListを記録するとともに、記録されているPlayListを管理する管理情報ファイルに、前記PlayListファイルに関する情報を追加する追加ステップとを含むことを特徴とするデータ処理方法。

【請求項22】 AVストリームファイル、および前記AVストリームの再生方法を指定するPlayListファイルが記録されている記録媒体から、データを受信するデータ処理装置のプログラムであって、

前記PlayListファイルを受信する受信ステップと、

前記PlayListを記録するとともに、記録されているPlayListを管理する管理情報ファイルに、前記PlayListファイルに関する情報を追加する追加ステップとを含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

【請求項23】 AVストリームファイル、および前記AVストリームの再生方法を指定するPlayListファイルが記録されている記録媒体から、データを受信するデータ処理装置を制御するコンピュータに、

前記PlayListファイルを受信する受信ステップと、

前記PlayListを記録するとともに、記録されているPlayListを管理する管理情報ファイルに、前記PlayListファイルに関する情報を追加する追加ステップとを実行させるプログラム。

【請求項24】 AVストリームファイル、および前記AVストリームのサムネールファイルが記録される記録媒体に対して、データを入力するデータ処理装置において、サムネールファイルを受信する受信部と、記録されているサムネールファイルに、受信したサムネールファイルのデータを追加する記録部とを備えることを特徴とするデータ処理装置。

【請求項25】 AVストリームファイル、および前記AVストリームのサムネールファイルが記録される記録媒体に対して、データを入力するデータ処理装置のデータ処理方法において、

サムネールファイルを受信する受信ステップと、

記録されているサムネールファイルに、受信したサムネールファイルのデータを追加する追加ステップとを含むことを特徴とするデータ処理方法。

【請求項26】 AVストリームファイル、および前記AVストリームのサムネールファイルが記録される記録媒体に対して、データを入力するデータ処理装置のプログラ

ムであつて、

サムネールファイルを受信する受信ステップと、記録されているサムネールファイルに、受信したサムネールファイルのデータを追加する追加ステップとを含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

【請求項27】 AVストリームファイル、および前記AVストリームのサムネールファイルが記録される記録媒体に対して、データを入力するデータ処理装置を制御するコンピュータに、

サムネールファイルを受信する受信ステップと、記録されているサムネールファイルに、受信したサムネールファイルのデータを追加する追加ステップとを実行させるプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明はデータ伝送装置および方法、データ処理装置および方法、記録媒体、並びにプログラムに関し、特に、記録媒体に記録されているデータの内容を編集した場合においても、記録媒体に記録されているデータ内容、および、再生情報を適切に管理することができるようとしたデータ伝送装置および方法、データ処理装置および方法、記録媒体、並びにプログラムに関する。

【0002】

【従来の技術】 近年、記録再生装置から取り外し可能なディスク型の情報記録媒体として、各種の光ディスクが提案されつつある。このような記録可能な光ディスクは、数ギガバイトの大容量メディアとして提案されており、ビデオ信号等のAV(Audio Visual)信号を記録するメディアとしての期待が高い。この記録可能な光ディスクに記録するデジタルのAV信号のソース(供給源)としては、CSデジタル衛星放送やBSデジタル放送があり、また、将来はデジタル方式の地上波テレビジョン放送等も提案されている。

【0003】 ここで、これらのソースから供給されるデジタルビデオ信号は、通常MPEG (Moving Picture Experts Group) 2方式で画像圧縮されているのが一般的である。また、記録装置には、その装置固有の記録レートが定められている。従来の民生用映像蓄積メディアで、デジタル放送からのデジタルビデオ信号を記録する場合、アナログ記録方式であれば、デジタルビデオ信号をデコード後、帯域制限をして記録が行われる。あるいは、MPEG 1 Video、MPEG 2 Video、DV (Digital Video) 方式をはじめとするデジタル記録方式であれば、一度デコードされた後に、その装置固有の記録レート、かつ符号化方式で再エンコードされて記録される。

【0004】 しかしながら、このような記録方法は、供給されたビットストリームを一度デコードし、その後で帯域制限や再エンコードを行って記録するため、画質の

劣化を伴う。画像圧縮されたデジタル信号の記録をする場合、入力されたデジタル信号の伝送レートが記録再生装置の記録レートを超えない場合には、供給されたビットストリームをデコードや再エンコードすることなく、そのまま記録する方法が最も画質の劣化が少ない。ただし、画像圧縮されたデジタル信号の伝送レートが記録媒体としてのディスクの記録レートを超える場合には、記録再生装置でデコード後、伝送レートがディスクの記録レートの上限以下になるように、再エンコードをして記録する必要がある。

【0005】 また、入力デジタル信号のビットレートが時間により増減する可変レート方式によって伝送されている場合には、回転ヘッドが固定回転数であるために記録レートが固定レートになるテープ記録方式に比べ、一度バッファにデータを蓄積し、バースト的に記録ができるディスク記録装置の方が、情報記録媒体としてのディスクの容量をより無駄なく利用できる。

【0006】 以上のように、デジタル放送が主流となる将来においては、データストリーマのように放送信号をデジタル信号のまま、デコードや再エンコードすることなく記録し、記録媒体としてディスクを使用した記録再生装置が求められると予測される。

【0007】

【発明が解決しようとする課題】 上述したように、記録媒体の容量が増大することにより、その記録媒体には、多くのデータ(例えば、番組に関する映像データや音声データなど)が記録できるようになり、1枚のディスクに多くの番組が記録されることになる。この時に、ユーザが、それらのディスク内に記録されている多くの番組の中から、所望のデータを別のディスクにコピーする、といったような操作が必要になる。

【0008】 しかしながら、コピー操作が行われた場合、記録されているデータの内容、および、再生情報を適切に管理することが困難になる。

【0009】 本発明はこのような状況に鑑みてなされたものであり、記録媒体に記録されているデータの内容を別の記録媒体にコピーした場合においても、記録媒体に記録されているデータの内容、および、再生情報を適切に管理することができるようすることを目的とする。

【0010】

【課題を解決するための手段】 本発明の第1のデータ伝送装置は、データストリームとその管理情報を記録媒体から読み出す読み出し部と、データストリームの中で、指定された再生区間の再生に必要な部分データストリームに対応する部分管理情報を決定する制御部と、部分データストリーム、および、制御部により決定された部分管理情報を伝送する伝送部とを備えることを特徴とする。

【0011】 前記伝送部は、さらに部分管理情報を、非同期伝送することができる。

【0012】前記制御部は、接続された機器の種別に応じて、部分データストリームを非同期伝送するか、または同期伝送するかを選択することができる。

【0013】前記データストリームは、AVストリームであり、管理情報は、AVストリーム中の符号化情報の不連続点のアドレス情報、AVストリーム中の時刻情報とアドレス情報を関連づける情報、およびAVストリーム中の特徴的な画像の時刻情報を含むようにすることができる。

【0014】前記管理情報は、Clip Informationであり、不連続点のアドレス情報は、SequenceInfoおよびProgramInfoであり、時刻情報とアドレス情報を関連づける情報は、CPIであり、特徴的な画像の時刻情報は、ClipMarkであるようにすることができる。

【0015】前記部分管理情報は、AVストリームの再生区間の指示情報である、PlayListをさらに含むようにすることができる。

【0016】前記伝送部は、部分データストリームとしてのAVデータストリームの再生区間の指示情報をさらに伝送することができる。

【0017】前記伝送部は、AVストリームの再生区間の指示情報の内容を変更しないで、部分データストリームとしてのAVデータストリームの再生区間の指示情報として伝送することができる。

【0018】前記伝送部は、AVストリームの再生区間の指示情報に関連付けられたサムネール画像、および、部分管理情報データに含まれるAVストリーム中の特徴的な画像の時刻情報に関連付けられたサムネール画像をさらに伝送することができる。

【0019】前記AVストリームは、トランSPORTパケットとそのアライバルタイムスタンプから構成されるソースパケットを単位とするデータ列であり、部分データストリームとしてのAVストリームは、AVストリームのソースパケットのデータ列の部分であるようにすることができる。

【0020】前記AVストリームは、トランSPORTパケットとそのアライバルタイムスタンプから構成されるソースパケットを単位とするデータ列であり、部分データストリームとしてのAVストリームは、トランSPORTパケットを単位とするトランSPORTストリームであるようにすることができる。

【0021】制御部は、AVストリームとその管理情報を共にデータ伝送するか、またはAVストリームだけをリアルタイム転送するかを切り替えることができる。

【0022】本発明の第1のデータ伝送方法は、データストリームとその管理情報を記録媒体から読み出す読み出しステップと、データストリームの中で、指定された再生区間の再生に必要な部分データストリームに対応する部分管理情報を決定する決定ステップと、部分データストリーム、および、決定ステップの処理により決定された管理情報を伝送する伝送ステップとを含むことを特

10

20

30

40

40

50

徴とする。

【0023】本発明の第1の記録媒体のプログラムは、データストリームとその管理情報を記録媒体から読み出す読み出しステップと、データストリームの中で、指定された再生区間の再生に必要な部分データストリームに対応する部分管理情報を決定する決定ステップと、部分データストリーム、および、決定ステップの処理により決定された管理情報を伝送する伝送ステップとを含むことを特徴とする。

【0024】本発明の第1のプログラムは、データストリームとその管理情報を記録媒体から読み出す読み出しステップと、データストリームの中で、指定された再生区間の再生に必要な部分データストリームに対応する部分管理情報を決定する決定ステップと、部分データストリーム、および、決定ステップの処理により決定された管理情報を伝送する伝送ステップとを含むことを特徴とする。

【0025】本発明の第2のデータ伝送装置は、AVストリームの伝送先の装置と相互認証する認証部と、認証部による相互認証の結果から、伝送先の装置が所定のフォーマットに準拠していることが判った場合、AVストリームとその管理情報を共にデータ伝送するとともに、認証部による相互認証の結果から、伝送先の装置が所定のフォーマットに準拠していないことが判った場合、AVストリームだけをリアルタイム伝送する伝送部とを備えることを特徴とする。

【0026】本発明の第2のデータ伝送方法は、AVストリームの伝送先の装置と相互認証する相互認証ステップと、認証ステップの処理による相互認証の結果から、伝送先の装置が所定のフォーマットに準拠していることが判った場合、AVストリームとその管理情報を共にデータ伝送するとともに、認証ステップの処理による相互認証の結果から、伝送先の装置が所定のフォーマットに準拠していないことが判った場合、AVストリームだけをリアルタイム伝送する伝送ステップとを含むことを特徴とする。

【0027】本発明の第2の記録媒体のプログラムは、AVストリームを伝送するデータ伝送装置のプログラムであって、AVストリームの伝送先の装置と相互認証する相互認証ステップと、認証ステップの処理による相互認証の結果から、伝送先の装置が所定のフォーマットに準拠していることが判った場合、AVストリームとその管理情報を共にデータ伝送するとともに、認証ステップの処理による相互認証の結果から、伝送先の装置が所定のフォーマットに準拠していないことが判った場合、AVストリームだけをリアルタイム伝送する伝送ステップとを含むことを特徴とする。

【0028】本発明の第2のプログラムは、AVストリームを伝送するデータ伝送装置を制御するコンピュータに、AVストリームの伝送先の装置と相互認証する相互認

証ステップと、認証ステップの処理による相互認証の結果から、伝送先の装置が所定のフォーマットに準拠していることが判った場合、AVストリームとその管理情報を共にデータ伝送するとともに、認証ステップの処理による相互認証の結果から、伝送先の装置が所定のフォーマットに準拠していないことが判った場合、AVストリームだけをリアルタイム伝送する伝送ステップとを実行させる。

【0029】本発明の第1のデータ処理装置は、PlayListファイルを受信する受信部と、PlayListを記録するとともに、記録されているPlayListを管理する管理情報ファイルに、PlayListファイルに関する情報を追加する記録部とを備えることを特徴とする。

【0030】本発明の第1のデータ処理方法は、PlayListファイルを受信する受信ステップと、PlayListを記録するとともに、記録されているPlayListを管理する管理情報ファイルに、PlayListファイルに関する情報を追加する追加ステップとを含むことを特徴とする。

【0031】本発明の第3の記録媒体のプログラムは、AVストリームファイル、およびAVストリームの再生方法を指定するPlayListファイルが記録されている記録媒体から、データを受信するデータ処理装置のプログラムであって、PlayListファイルを受信する受信ステップと、PlayListを記録するとともに、記録されているPlayListを管理する管理情報ファイルに、PlayListファイルに関する情報を追加する追加ステップとを含むことを特徴とする。

【0032】本発明の第3のプログラムは、AVストリームファイル、およびAVストリームの再生方法を指定するPlayListファイルが記録されている記録媒体から、データを受信するデータ処理装置を制御するコンピュータに、PlayListファイルを受信する受信ステップと、PlayListを記録するとともに、記録されているPlayListを管理する管理情報ファイルに、PlayListファイルに関する情報を追加する追加ステップとを実行させる。

【0033】本発明の第2のデータ処理装置は、サムネールファイルを受信する受信部と、記録されているサムネールファイルに、受信したサムネールファイルのデータを追加する記録部とを備えることを特徴とする。

【0034】本発明の第2のデータ処理方法は、サムネールファイルを受信する受信ステップと、記録されているサムネールファイルに、受信したサムネールファイルのデータを追加する追加ステップとを含むことを特徴とする。

【0035】本発明の第4の記録媒体のプログラムは、AVストリームファイル、およびAVストリームのサムネールファイルが記録される記録媒体に対して、データを入力するデータ処理装置のプログラムであって、サムネールファイルを受信する受信ステップと、記録されているサムネールファイルに、受信したサムネールファイルの

データを追加する追加ステップとを含むことを特徴とする。

【0036】本発明の第4のプログラムは、AVストリームファイル、およびAVストリームのサムネールファイルが記録される記録媒体に対して、データを入力するデータ処理装置を制御するコンピュータに、サムネールファイルを受信する受信ステップと、記録されているサムネールファイルに、受信したサムネールファイルのデータを追加する追加ステップとを実行させる。

【0037】本発明の第1のデータ伝送装置および方法、記録媒体のプログラム、並びにプログラムにおいては、データストリームの中で指定された再生区間に再生に必要な部分データストリームおよび部分管理情報データが決定され、決定された部分データストリームが伝送される。

【0038】本発明の第2のデータ伝送装置および方法、記録媒体のプログラム、並びにプログラムにおいては、伝送先の装置が所定のフォーマットに準拠している場合、AVストリームとその管理情報データがともにデータ伝送され、準拠していない場合、AVストリームだけがリアルタイム伝送される。

【0039】本発明の第1のデータ処理装置および方法、記録媒体のプログラム、並びにプログラムにおいては、PlayListファイルが管理情報ファイルに追加される。

【0040】本発明の第2のデータ処理装置および方法、記録媒体のプログラム、並びにプログラムにおいては、第1のサムネールファイルのデータが、第2のサムネールファイルに追加される。

【0041】

【発明の実施の形態】以下に、本発明の実施の形態について、図面を参照して説明する。

【0042】図1は、記録媒体（後述する図44の記録媒体10）上のアプリケーションフォーマットの簡単化された構造を示している。このフォーマットは、AVストリームの管理のためにPlayListとClipの2個のレイヤをもつ。そして、Volume Informationは、ディスク内のすべてのClipとPlayListの管理をする。

【0043】1個のAVストリームと、その付属情報のペアを1個のオブジェクトと考え、それをClipと呼ぶ。AVストリームファイルはClip AVストリームファイルと呼ばれ、その付属情報は、Clip Information fileと呼ばれる。

【0044】1個のClip AVストリームファイルは、MPE G2トランSPORTストリームをDVR (Digital Video Recording) アプリケーションフォーマットによって規定される構造に配置したデータをストアする。

【0045】一般に、コンピュータ等で用いるデータファイルは、バイト列として扱われるが、Clip AVストリームファイルのコンテンツは、時間軸上に展開され、P1

ayListは、Clipの中のアクセスポイントを主にタイムスタンプで指定する。PlayListによって、Clipの中のアクセスポイントのタイムスタンプが与えられた時、ClipInformation fileは、Clip AVストリームファイルの中でストリームのデコードを開始すべきアドレス情報を見つけるために役立つ。

【0046】PlayListは、Clipの中からユーザが見たい再生区間を選択し、それを簡単に編集することができる目的にして導入された。1つのPlayListは、Clipの中の再生区間の集まりである。あるClipの中の1つの再生区間は、PlayItemと呼ばれ、それは、時間軸上のIN点とOUT点のペアで表される。それゆえ、PlayListは、PlayItemの集まりである。

【0047】PlayListには、2つのタイプがある。1つは、Real PlayListであり、もう1つは、Virtual PlayListである。

【0048】Real PlayListは、それが参照しているClipのストリーム部分を共有しているとみなされる。すなわち、Real PlayListは、それが参照しているClipのストリーム部分に相当するデータ容量をディスクの中で占める。AVストリームが新しいClipとして記録される場合、そのClip全体の再生可能範囲を参照するReal PlayListが自動的に作られる。Real PlayListの再生範囲の一部分が消去された場合、それが参照しているClipのストリーム部分のデータもまた消去される。

【0049】Virtual PlayListは、Clipのデータを共有していないとみなされる。Virtual PlayListが変更または消去されたとしても、Clipは何も変化しない。

【0050】なお、以下の説明においては、Real PlayListとVirtual PlayListを総称して単に、PlayListと呼んでいる。

【0051】DVRディスク上に必要なディレクトリは、次の通りである。“DVR”ディレクトリを含むrootディレクトリ“PLAYLIST”ディレクトリ、“CLIPINF”ディレクトリ、“STREAM”ディレクトリおよび“DATA”ディレクトリを含む“DVR”ディレクトリ

【0052】rootディレクトリの下に、これら以外のディレクトリを作っても良いが、それらは、このDVRアプリケーションフォーマットでは、無視される。

【0053】図2に、DVRディスク上のディレクトリ構造の例を示す。同図に示されるように、rootディレクトリは、1個のディレクトリを含む。“DVR” — DVRアプリケーションフォーマットによって規定されるすべてのファイルとディレクトリは、このディレクトリの下にストアされなければならない。

【0054】“DVR”ディレクトリは、次に示すファイルをストアする。“info.dvr”ファイルは、DVRディレクトリの下に作られ、アプリケーションレイヤの全体的な情報をストアする。DVRディレクトリの下には、ただ一つのinfo.dvrがなければならない。ファイル名は、info.d

vrに固定されるとする。“menu.tidx”と“menu.tdat”、そして“mark.tidx”と“mark.tdat”は、サムネールについての情報をストアするためのファイルである。

【0055】“DVR”ディレクトリは、以下に説明するディレクトリを含む。“PLAYLIST” — Real PlayListとVirtual PlayListのデータベースファイルは、このディレクトリの下に置かなければならない。このディレクトリは、PlayListが1個もなくても存在しなければならない。“CLIPINF” — Clipのデータベースは、このディレクトリの下に置かなければならない。このディレクトリは、Clipが1個もなくても存在しなければならない。“STREAM” — AVストリームファイルは、このディレクトリの下に置かなければならない。このディレクトリは、AVストリームファイルが1個もなくても存在しなければならない。

【0056】“PLAYLIST”ディレクトリは、2種類のPlayListファイルをストアするものであり、それらは、Real PlayListとVirtual PlayListである。“xxxxx.rpls” — このファイルは、1個のReal PlayListに関連する情報をストアする。それぞれのReal PlayList毎に、1個のファイルが作られる。ファイル名は、“xxxxx.rpls”である。ここで、“xxxxx”は、5個の0から9までの数字である。ファイル拡張子は、“rpls”でなければならない。“yyyy.y.vpls” — このファイルは、1個のVirtual PlayListに関連する情報をストアする。それぞれのVirtual PlayList毎に、1個のファイルが作られる。ファイル名は、“yyyy.y.vpls”である。ここで、“yyyy”は、5個の0から9までの数字である。ファイル拡張子は、“vpls”でなければならない。

【0057】“CLIPINF”ディレクトリは、それぞれのAVストリームファイルに対応して、1個のファイルをストアする。“zzzzz.clpi” — このファイルは、1個のAVストリームファイル(Clip AVストリームファイルまたはBridge-Clip AVストリームファイル)に対応するClipInformation fileである。ファイル名は、“zzzzz.clpi”であり、ここで、“zzzzz”は、5個の0から9までの数字である。ファイル拡張子は、“clpi”でなければならない。

【0058】“STREAM”ディレクトリは、AVストリームのファイルをストアする。“zzzzz.m2ts” — このファイルは、DVRシステムにより扱われるAVストリームファイルである。これは、Clip AVストリームファイルまたはBridge-Clip AVストリームファイルである。ファイル名は、“zzzzz.m2ts”であり、ここで“zzzzz”は、5個の0から9までの数字である。ファイル拡張子は、“m2ts”でなければならない。

【0059】1個のAVストリームファイルとそれに対応するClip information fileは、同じ5個の数字“zzzzz”を使用しなければならない。

【0060】その他のディレクトリとファイル名は、本発明の実施の形態を説明するために必要ないので、説明

を省略する。

【0061】次に、AVストリームファイルの構造を説明する。AVストリームファイルは図3示すDVR MPEG2トランSPORTストリームの構造を持たなければならない。DVRMPEG2トランSPORTストリームは次に示す特徴を持つ。

【0062】DVR MPEG2トランSPORTストリームは、整数個のAligned unitから構成される。Aligned unitの大きさは、6144バイト(2048×3バイト)である。Aligned unitは、ソースパケットの第1バイト目から始まる。ソースパケットは、192バイト長である。1個のソースパケットは、TP_extra_headerとトランSPORTパケットから成る。TP_extra_headerは、4バイト長であり、またトランSPORTパケットは、188バイト長である。1個のAligned unitは、32個のソースパケットから成る。DVR MPEG2トランSPORTストリームの中の最後のAligned unitも、また32個のソースパケットから成る。最後のAligned unitが、入力トランSPORTストリームのトランSPORTパケットで完全に満たされなかつた場合、残りのバイト領域をヌルパケット(PID=0x1FFFのトランSPORTパケット)を持ったソースパケットで満たさねばならない。

【0063】Source packetのシンタクスを図4に示す。

【0064】TP_extra_header()は、4バイト長のヘッダである。また、transport_packet()は、ISO/IEC 13818-1で規定される188バイト長のMPEG-2トランSPORTパケットである。

【0065】TP_extra_headerのシンタクスを図5に示す。

【0066】copy_permission_indicatorは、対応するトランSPORTパケットのペイロードのコピー制限を表す整数である。

【0067】arrival_time_stampは、AVストリームの中で、対応するトランSPORTパケットがデコーダ(後述する図44のAVデコーダ16が対応する)に到着する時刻を示すタイムスタンプである。これは、後述する式(1)の中でarrival_time_stampによって指定される値を持つ整数値である。

【0068】図6は、DVR MPEG-2トランSPORTストリーム*40 arrival_time_stamp(k)

【0073】ライトバッファ(Write Buffer)205(図44の動画像記録再生装置1では、書き込み部32に内蔵されている)について説明する。Rmaxは、ソースパケットタイザ204からライトバッファ205へのソースパケットストリームの入力ビットレートである。入力トランSPORTストリームの最大ビットレートをTS_recording_rateとすると、Rmaxは次のように計算される。

$$R_{max} = TS_recording_rate \times 192/188$$

【0074】Rudは、ライトバッファ205からDVRドライ

*一ムのレコーダモデル(後述する図44の動画像記録再生装置1が対応する)を示す。これは、レコーディングプロセスを規定するための概念上のモデルである。DVR MPEG-2トランSPORTストリームは、このモデルに従つていなければならない。

【0069】MPEG-2トランSPORTストリームの入力タイミングについて説明する。入力MPEG2トランSPORTストリームは、フルトランSPORTストリームまたはパーシャルトランSPORTストリームである。入力MPEG2トランSPORTストリームは、ISO/IEC13818-1またはISO/IEC13818-9に従つていなければならない。MPEG2トランSPORTストリームのi番目のバイトは、T-STD(ISO/IEC 13818-1で規定されるTransport stream system target decoder)201(図44のAVデコーダ16が対応する)とソースパケットタイザ(source packetizer)204(図44のソースパケットタイザ29が対応する)へ、時刻t(i)に同時に入力される。

【0070】27MHz PLL202(図44の動画像記録再生装置1では、制御部17に内蔵されている)について説明する。27MHzクロックの周波数は、MPEG-2トランSPORTストリームのPCR(Program Clock Reference)の値にロックしなければならない。

【0071】arrival_time_clockについて説明する。アライバルタイムクロックカウンタ(Arrival time clock counter)203(図44の動画像記録再生装置1では、制御部17に内蔵されている)は、27MHz PLL202が输出する27MHzの周波数のパルスをカウントするバイナリーカウンターである。Arrival_time_clock(i)は、時刻t(i)におけるArrival time clock counter203のカウント値である。

【0072】ソースパケットタイザ204について説明する。ソースパケットタイザ204は、すべてのトランSPORTパケットにTP_extra_headerを付加し、ソースパケットを作る。Arrival_time_stampは、トランSPORTパケットの第1バイト目がT-STD201とソースパケットタイザ204の両方へ到着する時刻を表す。Arrival_time_stamp(k)は、等式(1)で示されるように、Arrival_time_clock(k)のサンプル値であり、ここで、kはトランSPORTパケットの第1バイト目を示す。

$$arrival_time_stamp(k) = arrival_time_clock(k) \% 2^{30} \quad (1)$$

イブ(DVR drive)206(図44の動画像記録再生装置1では、書き込み部32に内蔵されている)への出力ビットレートである。ライトバッファ205が空でない時のバッファからのソースパケットストリームの出力ビットレートはRudである。バッファが空である時、バッファからの出力ビットレートはゼロである。

【0075】DVRドライブ206は、T-STD201への各パケットの到着時刻に対応するATSが付加された、ライトバッファ205からの各パケットを、ディスク(図4

4の記録媒体10に対応する)に記録する。

【0076】図7は、DVR MPEG-2トランSPORTストリームのプレーヤモデル(図44の動画像記録再生装置1が対応する)を示す。これは、再生プロセスを規定するための概念上のモデルである。DVR MPEG-2トランSPORTストリームは、このモデルに従っていなければならない。

【0077】リードバッファ(Read buffer) 222
(図44の動画像記録再生装置1では、読み出し部11に内蔵されている)について説明する。Rudは、DVRドライブ(DVR drive) 221(図44の動画像記録再生装置1では、読み出し部11に内蔵されている)からリードバッファ222への入力ビットレートである。リードバッファ222がフルでない時のバッファへのソースパケットストリームの入力ビットレートはRudである。バッファがフルである時、バッファへの入力は止められる。Rmaxは、リードバッファ222からソースデパケットタイザ(source depacketizer) 223(図43のソースデパケットタイザ14が対応する)へのソースパケットストリームの出力ビットレートである。

【0078】アライバルタイムクロックカウンタ(arrival time clock counter) 225(図44の動画像記録再生装置1では、制御部17に内蔵されている)について説明する。アライバルタイムクロックカウンタ225は、27MHzクリスタル発振器(27MHzX-tal) 224(図44の動画像記録再生装置1では、制御部17に内蔵されている)が発生する27MHzの周波数のパルスをカウントするバイナリーカウンターである。現在のソースパケットがAVストリームファイルの最初のソースパケットであるか、または後述するSequenceInfo()の中でSPN_ATC_startが指すところのソースパケットである場合、そのパケットのarrival time stampの値でアライバルタイムクロックカウンタ225のカウント値をリセットする。Arrival_time_clock(i)は、時刻t(i)におけるアライバルタイムクロックカウンタ225のカウント値である。

【0079】MPEG-2トランSPORTストリームの出力タイミングについて説明する。現在のソースパケットのarrival_time_stampがarrival_time_clock(i)のLSB 30ビットの値と等しい時、そのソースパケットのトランSPORTパケットは、バッファから引き抜かれる。

【0080】次に、AVストリームファイルの再生情報を管理するデータベースフォーマットについて説明する。

【0081】図8は、Clip Information fileのシンタクスを示す。Clip Information fileは、SequenceInfo(), ProgramInfo(), CPI(), ClipMark()を持つ。

【0082】SequenceInfo_start_addressは、zzzzz.clpiファイルの先頭のバイトからの相対バイト数を単位として、SequenceInfo()の先頭アドレスを示す。相対バ

ト数はゼロからカウントされる。

【0083】ProgramInfo_Start_addressは、zzzzz.clpiファイルの先頭のバイトからの相対バイト数を単位として、ProgramInfo()の先頭アドレスを示す。相対バイト数はゼロからカウントされる。

【0084】CPI_Start_addressは、zzzzz.clpiファイルの先頭のバイトからの相対バイト数を単位として、CPI()の先頭アドレスを示す。相対バイト数はゼロからカウントされる。

【0085】ClipMark_Start_addressは、zzzzz.clpiファイルの先頭のバイトからの相対バイト数を単位として、ClipMark()の先頭アドレスを示す。相対バイト数はゼロからカウントされる。

【0086】その他のシンタクスフィールドは、本発明の実施の形態を説明するために必要ないので、説明を省略する。

【0087】SequenceInfo()は、Clip AV streamの中のATC-sequenceとSTC-sequenceの情報を定義する。

【0088】ATC-sequenceについて説明する。AVストリームファイルを構成する各ソースパケットのarrival time stamp(ATS)に基づいて作られる時間軸を、アライバルタイムベースと呼び、そのクロックを、ATC(Arrival Time Clock)と呼ぶ。そして、ATCの不連続点(アライバルタイムベースの不連続点)を含まないソースパケット列を、ATC-sequenceと呼ぶ。

【0089】図9は、ATC-sequenceについて説明する図である。入力トランSPORTストリームをClip AVストリームファイルとして新しく記録する時、そのClipはATCの不連続点を含んではならず、ただ1つのATC-sequenceを持つ。ATCの不連続点は、編集等によってClip AVストリームファイルのストリームデータを部分的に消去した場合にだけ作られることを想定している。これについての詳細は後述する。

【0090】AVストリームファイルの中で、新しいATCが開始するアドレス、すなわち、ATC-sequenceのスタートアドレスを、SequenceInfo()にストアする。このアドレスは、SPN_ATC_startにより示される。

【0091】AVストリームファイルの中にある最後のATC-sequence以外のATC-sequenceは、そのSPN_ATC_startで指されるソースパケットから開始し、その次のSPN_ATC_startで指されるソースパケットの直前のソースパケットで終了する。最後のATC-sequenceは、そのSPN_ATC_startで指されるソースパケットから開始し、AVストリームファイルの最後のソースパケットで終了する。

【0092】図10は、ATCの不連続点とATC-sequenceの関係を説明する図である。この例の場合、Clip AVストリームファイルは、2個のATC不連続点を持ち、3個のATC-sequenceを持つ。

【0093】STC-sequenceについて説明する。STC(System Time Clock)の定義は、MPEG-2で規定されている定義

に従う。すなわち、これは、トランSPORTストリームの中のPCR(Program Clock Reference)に基づいて作られる時間軸であるシステムタイムベースのクロックである。STCの値は90kHz精度、33ビット長のバイナリーカウンタのカウント値で表される。

【0094】図1.1は、連続なSTC区間について説明する図である。ここで横軸は、ArrivalTime Clock (またはアライバルタイムベース) であり、縦軸は STC (またはシステムタイムベース) である。Case-1の場合、STCは単調増加しており、その区間のSTCは連続である。Case-2の場合、33ビットのSTCが途中でラップアラウンドしている。STCのラップアラウンド点はSTCの不連続ではない。ラップアラウンドしてもSTCは連続である。

【0095】STCの不連続は、放送局が伝送系を切り替えた場合、記録側が記録するチャンネルを切り替えた場合、ユーザが編集動作を行った場合などに発生する。

【0096】STCの不連続点 (システムタイムベースの不連続点) を含まないソースパケット列を、STC-sequenceと呼ぶ。なお、同じSTC-sequenceの中で同じSTCの値は、決して現れない。そのために、Clipの最大時間長を、33ビットのSTCのラップアラウンド周期 (約26時間) 以下に制限している。

【0097】AVストリームファイルの中で、新しいSTCが開始するアドレス、すなわち、STC-sequenceのスタートアドレスが、SequenceInfo()にストアされる。このアドレスは、SPN_STC_startにより示される。

【0098】STC-sequenceは、ATC-sequenceの境界をまたぐことはない。

【0099】AVストリームファイルの中にある最後のSTC-sequence以外のSTC-sequenceは、そのSPN_STC_startで指されるソースパケットから開始し、その次のSPN_STC_startで指されるソースパケットの直前のソースパケットで終了する。最後のSTC-sequenceは、そのSPN_STC_startで指されるソースパケットから開始し、AVストリームファイルの最後のソースパケットで終了する。

【0100】図1.2は、STCの不連続点とSTC-sequenceの関係、およびSTC-sequenceとATC-sequenceの関係を説明する図である。この例の場合、Clip AVストリームファイルは、3個のSTCを持ち、3個のSTC-sequenceを持つ。1つのSTC-sequenceが、ATC-sequenceの境界をまたぐことはない。

【0101】AVストリームがSTCの不連続点を持つ場合、そのAVストリームファイルの中で同じ値のPTSが現れるかもしれない。そのため、AVストリーム上のある時刻をPTSベースで指す場合、アクセスポイントのPTSだけではそのポイントを特定するためには不十分である。PTSに加えて、そのPTSを含むところのSTC-sequenceのインデックスが必要である。そのインデックスをSTC-idと呼ぶ。

【0102】図1.3は、SequenceInfo()のシンタクスを

示す。

【0103】lengthは、このlengthフィールドの直後のバイトからSequenceInfo()の最後のバイトまでのバイト数を示す。

【0104】num_of_ATC_sequencesは、AVストリームファイルの中にあるATC-sequenceの数を示す。

【0105】SPN_ATC_start[atc_id]は、AVストリームファイル上でatc_idによって指されるATC-sequenceが開始するアドレスを示す。SPN_ATC_start[atc_id]は、ソースパケット番号を単位とする大きさであり、AVストリームファイルの最初のソースパケットからゼロを初期値としてカウントされる。

【0106】SequenceInfo()の中の最初のSPN_ATC_start[0]は、ゼロである。また、SequenceInfo()の中でエントリーされるSPN_ATC_start[atc_id]の値は、昇順に並んでいる。すなわち、SequenceInfo()の中でエントリーされるSPN_ATC_start[atc_id]は、次の条件を満たす。

【0107】 $SPN_ATC_start[0] = 0$
 $0 < atc_id < num_of_ATC_sequences$ なる atc_id について、

$SPN_ATC_start[atc_id - 1] < SPN_ATC_start[atc_id]$

【0108】num_of_STC_sequences[atc_id]は、atc_idによって指されるATC-sequence上にあるSTC-sequenceの数を示す。

【0109】offset_STC_id[atc_id]は、atc_idによって指されるATC-sequence上にある最初のSTC-sequenceに対するsct_id のオフセット値を示す。AVストリームファイルを新たに記録する時、offset_STC_id[atc_id]は、ゼロである。

【0110】atc_idによって指されるATC-sequence上にあるSTC-sequenceに対するsct_idの値は、シンタクス中のsct_idのfor-loopによって記述される順番によって定義され、その値はoffset_STC_id[atc_id]から開始する。

【0111】SequenceInfo()の中で定義される連続する2個のATC-sequenceについて、前側のATC-sequenceの最後の sct_id と それに続くATC-sequenceの最初の sct_idは、同じ値でも良い。もし、これら2個の sct_id が同じ値の場合、それらの値で参照される2個の STC-sequenceの中で同じSTCの値が現れることはない。

【0112】SequenceInfo()の中でエントリーされるsct_idの値は、昇順に並ばなければならない。offset_STC_id[atc_id]は、この制限を満たすように値がセットされる。

【0113】PCR_PID[atc_id][sct_id]は、atc_idによって指されるATC-sequence上にあるところのsct_idによって指されるSTC-sequence に有効なPCRを持つトランSPORTパケットのPIDの値である。

【0114】SPN_STC_start[atc_id][sct_id]は、atc_idによって指されるATC-sequence上にあるところのsct_id

dによって指されるSTC-sequenceが、AVストリームファイル上で開始するアドレスを示す。SPN_STC_start[atc_id][stc_id]は、ソースパケット番号を単位とする大きさであり、AVストリームファイルの最初のソースパケットからゼロを初期値としてカウントされる。

【0115】SequenceInfo()の中でエントリーされるSPN_STC_start[atc_id][stc_id]の値は、昇順に並んでいます。atc_idによって指されるATC-sequence上にある最初のSPN_STC_start[atc_id][stc_id]は、SPN_ATC_start[atc_id]以上の値である。すなわち、次の条件を満たす。

【0116】 $SPN_ATC_start[atc_id] \leq SPN_STC_start[atc_id][0]$

【0117】presentation_start_time[atc_id][stc_id]は、atc_idによって指されるATC-sequence上にあるstc_idによって指されるSTC-sequence上にあるAVストリームデータのプレゼンテーション・スタート・タイムを示す。これは、そのSTC-sequenceのSTCから導かれる45 kHzを単位とするプレゼンテーション・タイムの値である。

【0118】presentation_end_time[atc_id][stc_id]は、atc_idによって指されるATC-sequence上にあるstc_idによって指されるSTC-sequence上にあるAVストリームデータのプレゼンテーション・エンド・タイムを示す。これは、そのSTC-sequenceのSTCから導かれる45 kHzを単位とするプレゼンテーション・タイムの値である。

【0119】次に、ProgramInfo()について説明する。プログラムは、エレメンタリストリームの集まりであり、これらのストリームの同期再生のために、ただ1つのシステムタイムベースを共有するものである。

【0120】再生装置（後述する図44の動画像記録再生装置1）にとって、AVストリームのデコードに先だち、そのAVストリームの内容がわかることは有用である。この内容とは、例えば、ビデオやオーディオのエレメンタリーストリームを伝送するトランスポートパケットのPIDの値や、ビデオやオーディオのコンポーネント種類（例えば、HDTVのビデオとMPEG-2 AACのオーディオストリームなど）などの情報である。

【0121】この情報はAVストリームを参照するところのPlayListの内容をユーザーに説明するところのメニュー画面を作成するのに有用であるし、また、AVストリームのデコードに先だって、再生装置のAVデコーダ16

（後述する図44）およびデマルチプレクサ15（後述する図44）の初期状態をセットするために役立つ。この理由のために、Clip Information fileは、プログラムの内容を説明するためのProgramInfoを持つ。

【0122】MPEG2トランSPORTストリームをストアしているAVストリームファイルは、ファイルの中でプログラム内容が変化するかもしれない。例えば、ビデオエレメンタリーストリームを伝送するところのトランSPORTパケットのPIDが変化したり、ビデオストリームの

コンポーネント種類がSDTVからHDTVに変化するなどである。ProgramInfoは、AVストリームファイルの中でのプログラム内容の変化点の情報をストアする。

【0123】AVストリームファイルの中で本フォーマットが規定するプログラム内容が一定であるソースパケット列を、program-sequenceと呼ぶ。

【0124】AVストリームファイルの中で、新しいprogram-sequenceが開始するアドレスをProgramInfo()にストアする。このアドレスは、SPN_program_sequence_startにより示される。

【0125】AVストリームファイルの中にある最後のprogram-sequence以外のprogram-sequenceは、そのSPN_program_sequence_startで指されるソースパケットから開始し、その次のSPN_program_sequence_startで指されるソースパケットの直前のソースパケットで終了する。最後のprogram_sequenceは、そのSPN_program_sequence_startで指されるソースパケットから開始し、AVストリームファイルの最後のソースパケットで終了する。

【0126】図14は、program-sequenceを説明する図である。この例の場合、Clip AVストリームファイルは3個のprogram-sequenceを持つ。

【0127】program-sequenceは、ATC-sequenceの境界およびSTC-sequenceの境界をまたいでも良い。

【0128】図15は、ProgramInfo()のシンタクスを示す。

【0129】lengthは、このlengthフィールドの直後のバイトからProgramInfo()の最後のバイトまでのバイト数を示す。

【0130】num_of_program_sequencesは、AVストリームファイルの中にあるprogram-sequenceの数を示す。

【0131】SPN_program_sequence_startは、AVストリームファイル上でprogram-sequenceが開始するアドレスを示す。SPN_program_sequence_startは、ソースパケット番号を単位とする大きさであり、AVストリームファイルの最初のソースパケットから、ゼロを初期値としてカウントされる。ProgramInfo()の中でエントリーされるSPN_program_sequence_startの値は、昇順に並んでいます。

【0132】SPN_program_sequence_startは、そのprogram_sequenceに対する最初のPMTを持つソースパケットを指していることを前提とする。SPN_program_sequence_startは、データを記録する記録機（図44の動画像記録再生装置1が対応する）がトランSPORTストリーム中のPSI/SIを解析することによって作られる。記録機

（例えば、図44のビデオ解析部24または多重化ストリーム解析部26）がPSI/SIを解析し、その変化を検出するまでの遅延時間が必要なために、SPN_program_sequence_startは、実際のPSI/SIの変化点から所定の時間以内にあるソースパケットを指しても良い。

【0133】program_map_PIDは、そのprogram-sequence

eに適用できるPMT(program map table)を持つトランスポートパケットのPIDの値である。

【0134】num_of_streams_in_psは、そのprogram-sequenceの中で定義されるエレメンタリーストリームの数を示す。

【0135】num_of_groupsは、そのprogram-sequenceの中で定義されるエレメンタリーストリームのグループの数を示す。num_of_groupsは、1以上の値である。トランスポートストリームのPSI/SIがエレメンタリーストリームのグループ情報を持つ場合、num_of_groupsは、1以上の値をとることを想定している。それぞれのグループは、マルチ・ビュー・プログラム中の1つのビューを構成する。

【0136】stream_PIDは、そのprogram-sequenceのprogram_map_PIDが参照するところのPMTの中で定義されているエレメンタリーストリームに対するPIDの値を示す。

【0137】StreamCodingInfo()は、前記stream_PIDで指されるエレメンタリーストリームの情報を示す。詳細は後述する。

【0138】num_of_streams_in_groupは、エレメンタリーストリームのグループが持つエレメンタリーストリームの数を示す。

【0139】stream_indexは、前記エレメンタリーストリームのグループが持つエレメンタリーストリームに対応するところの、シンタクス中のfor-loopで定義されるstream_indexの値を示す。

【0140】図16は、StreamCodingInfo()のシンタクスを示す。

【0141】lengthは、このlengthフィールドの直後のバイトからStreamCodingInfo()の最後のバイトまでのバイト数を示す。

【0142】stream_coding_typeは、このStreamCodingInfo()に対応するstream_PIDで指されるエレメンタリーストリームの符号化タイプを示す。値の意味を図17に示す。

【0143】video_formatは、このStreamCodingInfo()に対応するstream_PIDで指されるビデオストリームのビデオフォーマットを示す。値の意味を図18に示す。

【0144】frame_rateは、このStreamCodingInfo()に対応するstream_PIDで指されるビデオストリームのフレームレートを示す。値の意味を図19に示す。

【0145】display_aspect_ratioは、このStreamCodingInfo()に対応するstream_PIDで指されるビデオストリームのディスプレイ・アスペクト・レシオを示す。値の意味を図20に示す。

【0146】cc_flagは、このStreamCodingInfo()に対応するstream_PIDで指されるビデオストリームの中でクローズド・キャプション(closed caption data)信号が符号化されているかを示すフラグである。

【0147】original_video_format_flagは、このStreamCodingInfo()の中にoriginal_video_formatとoriginal_display_aspect_ratioが存在するかを示すフラグである。

【0148】original_video_formatは、このStreamCodingInfo()に対応するstream_PIDで指されるビデオストリームが符号化される前のオリジナルのビデオフォーマットである。値の意味は、前記のvideo_formatと同じである。

【0149】original_display_aspect_ratioは、このStreamCodingInfo()に対応するstream_PIDで指されるビデオストリームが符号化される前のオリジナルのディスプレイ・アスペクト・レシオである。値の意味は、前記のdisplay_aspect_ratioと同じである。

【0150】ビデオストリームと共にマルチメディアデータストリーム(BMLストリーム、字幕など)が多重化されているトランスポートストリームをトランス・コーディングする場合において、ビデオストリームは再エンコードされることによって、そのビデオフォーマットが変化する(例えば、1080iから480iへ変化する)が、マルチメディアデータストリームはオリジナルのストリームを保つ場合を考える。

【0151】この時、新しいビデオストリームとマルチメディアデータストリームの間に情報のミスマッチが生じる場合がある。例えば、マルチメディアデータストリームの表示に関するパラメータは、オリジナルのビデオストリームのビデオフォーマットを想定して決められているにもかかわらず、ビデオストリームの再エンコードによって、そのビデオフォーマットが変化した場合である。

【0152】このような場合、original_video_formatとoriginal_display_aspect_ratioに、オリジナルのビデオストリームに関する情報を保存する。再生機は、前記の新しいビデオストリームとマルチメディアデータストリームから次のようにして、表示画像をつくる。

【0153】ビデオストリームは、original_video_formatとoriginal_display_aspect_ratioで示されるビデオフォーマットにアップ・サンプリングされる。そのアップ・サンプリングされた画像とマルチメディアデータストリームが合成されて、正しい表示画像をつくる。

【0154】audio_presentation_typeは、このStreamCodingInfo()に対応するstream_PIDで指されるオーディオストリームのプレゼンテーション・タイプを示す。値の意味を図21に示す。

【0155】sampling_frequencyは、このStreamCodingInfo()に対応するstream_PIDで指されるオーディオストリームのサンプリング周波数を示す。値の意味を図22に示す。

【0156】次に、CPI()について説明する。CPI (Characteristic Point Information)は、AVストリームの中

の再生時間情報とそのファイルの中のアドレスとを関連づけるためにある。

【0157】CPIには2個のタイプがあり、それらはEP_mapとTU_mapである。CPI()の中のCPI_typeがEP_map typeの場合、そのCPI()はEP_mapを含む。また、CPI()の中のCPI_typeがTU_map typeの場合、そのCPI()はTU_mapを含む。1個のAVストリームファイルは、1個のEP_mapまたは1個のTU_mapを持つ。

【0158】EP_mapは、エントリーポイント(EP)データのリストであり、それはエレメンタリーストリームおよびトランスポートストリームから抽出されたものである。これは、AVストリームの中でデコードを開始すべきエントリーポイントの場所を見つけるためのアドレス情報を持つ。1つのEPデータは、プレゼンテーションタイムスタンプ(PTS)と、そのPTSに対応するアクセスユニットのAVストリームの中のデータアドレスの対で構成される。

【0159】EP_mapは、主に2つの目的のために使用される。第1に、PlayListの中でプレゼンテーションタイムスタンプによって参照されるアクセスユニットのAVストリームの中のデータアドレスを見つけるために使用される。第2に、ファーストフォワード再生やファーストリバース再生のために使用される。記録装置が、入力AVストリームを記録する場合、そのストリームのシンタクスを解析することができると、EP_mapが作成され、ディスクに記録される。

【0160】TU_mapは、デジタルインターフェースを通して入力されるトランスポートパケットの到着時刻に基づいたタイムユニット(TU)データのリストを持つ。これは、到着時刻ベースの時間とAVストリームの中のデータアドレスとの関係を与える。記録装置が、入力AVストリームを記録する場合、そのストリームのシンタクスを解析することができないと、TU_mapが作成され、ディスクに記録される。

【0161】図23はCPI()のシンタクスを示す。

【0162】lengthは、このlengthフィールドの直後のバイトからCPI()の最後のバイトまでのバイト数を示す。

【0163】CPI_typeは、1ビットのフラグであり、ClipのCPIのタイプを表す。

【0164】EP_mapは、AVストリームファイルの中にある1つのビデオストリームに対して、次に示すデータを持つ。

- (1) stream_PID : そのビデオストリームを伝送するトランスポートパケットのPIDを示す。
- (2) num_EP_entries : そのビデオストリームに対するエントリーポイントの数。EP_mapは、num_EP_entriesの数のPTS_EP_startとSPN_EP_startのペアのデータを持つ。
- (3) PTS_EP_start : そのビデオストリームの中で、シ

一ケンスヘッダから始まるアクセスユニットのPTSを示す。

(4) SPN_EP_start : 前記PTS_EP_startにより参照されるアクセスユニットの第1バイト目を含むソースパケットのAVストリームファイルの中でのアドレスを示す。SPN_EP_startはソースパケット番号を単位とする大きさであり、AVストリームファイルの最初のソースパケットから、ゼロを初期値としてカウントされる。

【0165】AVストリームファイルの中に複数のビデオストリームが存在する場合、EP_mapは各ビデオストリームに対して、前記のデータを持つことができる。

【0166】図24は、EP_mapの例を示す。ここでは、Clip AV streamの中に、stream_PID=xのビデオストリームがあり、k個のエントリーポイントがある (num_EP_entries=k)。SPN_EP_startで指されるソースパケットの例を図に示す。そのソースパケットの中のトランスポートパケットのTP_headerに続くペイロードは、PESパケットヘッダから開始する。それに続いて、シーケンスヘッダ(SQH)があり、それに続いてGOPヘッダ(GOPH)があり、それに続いてI-ピクチャヘッダ(I-PICH)がある。このシーケンスヘッダから始まるアクセスユニットのPTSは、PESパケットヘッダの中に符号化されている。

【0167】次に、TU_mapについて説明する。

【0168】図25は、AVストリームを新しくClipとして記録する時にできるTU_mapについて説明する図である。1つのATC-sequenceの中にあるソースパケットのアライバルタイムに基づいて作られる時間軸を所定の時間単位で分割する。この時間単位をtime-unitと呼ぶ。

【0169】各々のtime-unitの中に入るところの最初の完全な形のソースパケットのAVストリームファイル上でのアドレスをTU_mapにストアする。これらのアドレスをSPN_time_unit_startと呼ぶ。ATC-sequenceの上の時刻は、TU_start_timeに基づいて定義される。これについてSPN_time_unit_startのセマンティクスで後述する。

【0170】図26は、TU_mapのシンタクスを説明する図である。

【0171】time_unit_sizeは、1つのtime-unitの大きさを与えるものであり、それは27MHz精度のアライバルタイムクロックから導き出される45kHzクロックを単位とする大きさである。

【0172】シンタクス中のatc_idのfor-loopで使われているnum_of_ATC_sequencesの値は、SequenceInfo()の中で定義されている。

【0173】offset_arrival_time[atc_id]は、atc_idで指されるATC-sequenceの中の最初の完全なtime-unitに対するオフセットの時間である。これは、27MHz精度のアライバルタイムクロックから導き出される45kHzクロックを単位とする大きさである。

【0174】AVストリームを新しくClipとして記録した

時、そのAVストリームファイルはただ1つのATC-sequenceを持ち、offset_arrival_time[atc_id]はゼロである。

【0175】複数のoffset_arrival_time[atc_id]がTU_mapにエントリーされる場合は、次の条件式が満たされる。

```
offset_arrival_time[0] = 0
0 < atc_id < num_of_ATC_sequences なる atc_idについて、
offset_arrival_time[atc_id] > offset_arrival_time[atc_id-1]+time_unit*num_of_time_unit_entries[atc_id-1]
```

【0176】num_of_time_unit_entries[atc_id]は、atc_idで指されるATC-sequenceの中に含まれるtime_unitのエントリー数を示す。

【0177】SPN_time_unit_start[atc_id][i]は、atc_idで指されるATC-sequence中のi番目のtime_unitの開始するアドレスである。これはソースパケット番号を単位とし、AVストリームファイルの最初のソースパケットからゼロを初期値としてカウントされる。

【0178】現在のtime_unitに入るソースパケットが何もない場合、現在のtime_unitに対するSPN_time_unit_startの値は、その1つ前のSPN_time_unit_startの値と等しい。

【0179】TU_mapの中のSPN_time_unit_startの値のエントリは、昇順にならんでいなければならない。

【0180】atc_idで指されるATC-sequence中のi番目のtime_unitの開始時刻は、次式で定義されるTU_start_time[atc_id][i]である。

```
TU_start_time[atc_id][i]=offset_arrival_time[atc_id]+i*time_unit_size
```

【0181】次に、ClipMark()について説明する。

【0182】ClipMark()は、Clip AVストリームの中のマーク (Mark) の情報を定義する。マークは、Clipの中のハイライトや特徴的な時間を指定するために設けられている。Clipに付加されるマークは、AVストリームの内容に起因する特徴的なシーンを指定する、例えば、CM開始点やCM終了点、またシーンチェンジ点などである。Clipにセットされるマークは、AVストリームが新しいClipとして記録される時に記録器がセットする。なお、後でPlaylistファイルの中で説明するが、Playlistにセットされるマークをストアするために、PlaylistファイルはPlaylistMarkを持つ。Playlistにセットされるマークは、主にユーザによってセットされる、例えば、ブックマークやリジューム点などである。

【0183】ClipまたはPlaylistにマークをセットすることは、マークの時刻を示すタイムスタンプをClipMark / PlaylistMarkに追加することにより行われる。また、マークを削除することは、ClipMark / PlaylistMarkの中から、そのマークのタイムスタンプを削除する事

で行われる。従って、マークのセットや削除により、AVストリームは何の変更もされない。

【0184】図27は、ClipにセットされるマークとPlaylistにセットされるマークの関係を示す図である。Playlistを再生する時、そのPlaylistが参照するClipのClipMarkにストアされているマークを参照する事ができる。したがって、1つのClipをReal Playlistや複数のVirtual Playlistによって参照している場合、それらのPlaylistは、その1つのClipのClipMarkを共有することができる、マークのデータを効率良く管理することができる。

【0185】図28は、ClipMarkのシンタクスを示す。lengthは、このlengthフィールドの直後のバイトからClipMark()の最後のバイトまでのバイト数を示す。maker_IDは、そのmark_typeを定義しているメーカーのメーカーIDを示す。number_of_Clip_marksは、ClipMarkの中にストアされているマークのエントリー数を示す。

【0186】mark_invalid_flagは、1ビットのフラグであり、この値がゼロにセットされている時、このマークは有効な情報を持っていることを示し、また、この値が1にセットされている時、このマークは無効であることを示す。ユーザーがユーザーインターフェース上で1つのマークのエントリーを消去するオペレーションをした時、記録機はClipMarkからそのマークのエントリーを消去する代わりに、そのmark_invalid_flagの値を1に変更しても良い。mark_typeは、マークのタイプを示す。ref_to_STC_idは、mark_time_stampとrepresentative_picture_time_stampの両方が置かれているところのSTC-sequenceを指定するところのstc_idを示す。stc_idの値は、SequenceInfo()の中で定義される。

【0187】mark_time_stampは、Clip AVストリームの中でマークが指定されたポイントをプレゼンテーションタイムスタンプをベースとして表す。entry_ES_PIDが、0xFFFFにセットされている場合、そのマークはClipの中のすべてのエレメンタリーストリームに共通の時間軸へのポインターである。entry_ES_PIDが、0xFFFFでない値にセットされている場合、entry_ES_PIDは、そのマークによって指されるところのエレメンタリーストリームを含んでいるところのトランスポートパケットのPIDの値を示す。ref_to_thumbnail_indexは、マークに付加されるサムネール画像の情報を示す。ref_to_thumbnail_indexフィールドが、0xFFFFでない値の場合、そのマークにはサムネール画像が付加されており、そのサムネール画像は、mark.tdatファイルの中にストアされている。

その画像は、mark.tidxファイルの中でthumbnail_indexの値を用いて参照される（後述）。ref_to_thumbnail_indexフィールドが、0xFFFFである場合、そのマークにはサムネール画像が付加されていない事を示す。representative_picture_time_stampは、mark_time_stampによって示されるマークを代表する画像のポイントを示すタ

イムスタンプをストアする。

【0188】図29は、PlayList fileのシンタクスを示す。PlayList fileは、UIAppInfoPlayList(), PlayList(), PlayListMark()を持つ。

【0189】PlayList_start_addressは、PlayListファイルの先頭のバイトからの相対バイト数を単位として、PlayList()の先頭アドレスを示す。相対バイト数はゼロからカウントされる。PlayListMark_start_addressは、PlayListファイルの先頭のバイトからの相対バイト数を単位として、PlayListMark()の先頭アドレスを示す。相対バイト数はゼロからカウントされる。

【0190】他のシンタクスフィールドは、本発明の実施の形態を説明するために必要ないので、説明を省略する。

【0191】UIAppInfoPlayList()は、PlayListについてのユーザインターフェースアプリケーションのパラメータをストアする。

【0192】図30に、UIAppInfoPlayList()のシンタクスを示す。ref_to_thumbnail_indexは、マークに付加されるサムネール画像の情報を示す。ref_to_thumbnail_indexフィールドが、0xFFFFでない値の場合、そのマークにはサムネール画像が付加されており、そのサムネール画像は、menu.tdatファイルの中にストアされている。その画像は、menu.tidxファイルの中でthumbnail_indexの値を用いて参照される（後述）。ref_to_thumbnail_indexフィールドが、0xFFFFである場合、そのマークにはサムネール画像が付加されていない事を示す。他のシンタクスフィールドは、PlayListについてのユーザインターフェースアプリケーションのパラメータを示すものであるが、本発明の実施の形態を説明するために必要ないので、詳細な説明を省略する。

【0193】図31は、PlayList()のシンタクスを示す。

【0194】lengthは、このlengthフィールドの直後のバイトからPlayList()の最後のバイトまでのバイト数を示す。

【0195】CPI_type: 1ビットのフラグであり、PlayItem()が使用するClipのCPI_typeの値を示す。CPI_typeは、Clip Information fileのCPI_typeで定義される。

【0196】number_of_PlayItemsは、PlayList()の中にあるPlayItem()の数を示す。

【0197】シンタクス中のPlayItem_idのfor-loopの中で、PlayItem()の現れる順番によって、そのPlayItem()に対するPlayItem_idの値が決る。PlayItem_idは、0から開始する。

【0198】他のシンタクスフィールドは、本発明の実施の形態を説明するために必要ないので、説明を省略する。

【0199】次に、PlayItemについて説明する。1個のPlayItemは、基本的に次のデータを含む。

(1)PlayItem が指すClipのファイル名を指定するためのClip_information_file_name。

(2)そのClipの再生区間を特定するためのIN_timeとOUT_timeのペア。

(3)PlayListの中で連続する2個のPlayItemについて、前側のPlayItemと現在のPlayItemとの接続の状態を示すところのconnection_condition。

【0200】図32は、CPI_typeがEP_mapのPlayList（これをEP_map typeのPlayListと呼ぶ）を説明する図である。EP_map typeのPlayListの場合、PlayItemのIN_timeとOUT_timeは、PTSベースの時間を示す。そのIN_timeとOUT_timeは、同じSTC-sequence上の時刻を指す。そのSTC-sequenceを示すために、ref_to_STC_idを用いる。そのIN_timeとOUT_timeは、そのSTC-sequenceに対して定義される presentation_start_time と presentation_end_time（これらの情報はSequenceInfoの中にある）で示される再生区間の中の時間を指す。

【0201】図33は、CPI_typeがTU_mapのPlayList（これをTU_map typeのPlayListと呼ぶ）を説明する図である。TU_map typeのPlayListの場合、PlayItemのIN_timeとOUT_timeは、アライバルタイムベースの時間を指す。そのIN_timeとOUT_timeは、同じATC-sequence上の時刻を指す。

【0202】図34は、EP_map typeのPlayList の時間情報とAVストリームファイルの中のアドレス情報との関係を説明する図である。PlayList の時間情報は、AVストリームファイルの中のピクチャやオーディオフレームのPTS情報である。そして、Clip Information fileのEP_mapとSequenceInfoが、AVストリームの中の時間情報とそのファイルの中のアドレスとを関連づける。

【0203】図35は、TU_map typeのPlayList の時間情報とAVストリームファイルの中のアドレス情報との関係を説明する図である。PlayList の時間情報は、AVストリームファイルの中のアライバル・タイム情報である。そして、Clip Information fileのTU_mapが、AVストリームの中の時間情報とそのファイルの中のアドレスとを関連づける。

【0204】図36は、PlayItem()のシンタクスを示す。

【0205】lengthは、このlengthフィールドの直後のバイトからPlayItem()の最後のバイトまでのバイト数を示す。

【0206】Clip_Information_file_nameは、PlayItemが参照するClip Information fileのファイル名を示す。

【0207】connection_conditionは、先行するPlayItemと現在のPlayItemとがシームレスに接続されているかどうかの情報を示す。

【0208】ref_to_STC_idは、PlayItemが参照するClipのSTC-sequenceのstc-idを示す。stc-idの値は、SequenceInfoの中にある。

nceInfoの中で定義されている。

【0209】IN_timeは、PlayItemの再生開始時刻をストアする。

【0210】OUT_timeは、PlayItemの再生終了時刻をストアする。

【0211】Bridge_Clip_Information_file_nameは、先行するPlayItemと現在のPlayItemとがシームレスに接続されている場合の再生の補助情報である。

【0212】図37は、PlayListMark()のシンタクスを示す。PlayListMarkは前述したように、主にユーザによってセットされるマークをストアする。lengthは、このlengthフィールドの直後のバイトからPlayListMark()の最後のバイトまでのバイト数を示す。number_of_PlayList_marksは、PlayListMarkの中にストアされているマークのエントリー数を示す。mark_invalid_flagは、1ビットのフラグであり、この値がゼロにセットされている時、このマークは有効な情報を持っていることを示し、また、この値が1にセットされている時、このマークは無効であることを示す。ユーザーがユーザーインターフェース上で1つのマークのエントリーを消去するオペレーションをした時、記録機はPlayListMarkからそのマークのエントリーを消去する代わりに、そのmark_invalid_flagの値を1に変更しても良い。mark_typeは、マークのタイプを示す。mark_name_lengthは、Mark_nameフィールドの中に示されるマーク名のバイト長を示す。このフィールドの値は32以下である。

【0213】ref_to_PlayItem_idは、マークが置かれているところのPlayItemを指定するところのPlayItem_idの値を示す。あるPlayItemに対応するPlayItem_idの値は、PlayList()において定義される。mark_time_stampは、そのマークが指定されたポイントを示すタイムスタンプをストアする。mark_time_stampは、ref_to_PlayItem_idで示されるPlayItemの中で定義されているところのIN_timeとOUT_timeで特定される再生範囲の中の時間を指す。PlayListのCPI_typeがEP_map_typeの場合、mark_time_stampは、プレゼンテーションタイムスタンプをベースとして表され、また、CPI_typeがTU_map_typeの場合、mark_time_stampはアライバルタイムスタンプをベースとして表される。

【0214】entry_ES_PIDが、0xFFFFにセットされている場合、そのマークはPlayListによって使用されるすべてのエレメンタリーストリームに共通の時間軸上へのポインターである。entry_ES_PIDが、0xFFFFでない値にセットされている場合、entry_ES_PIDは、そのマークによって指されるところのエレメンタリーストリームを含んでいるところのトランスポートパケットのPIDの値を示す。ref_to_thumbnail_indexは、マークに付加されるサムネール画像の情報を示す。ref_to_thumbnail_indexフィールドが、0xFFFFでない値の場合、そのマークにはサムネール画像が付加されており、そのサムネール画像

は、mark.tdatファイルの中にストアされている。その画像は、mark.tidxファイルの中でthumbnail_indexの値を用いて参照される（後述）。ref_to_thumbnail_indexフィールドが、0xFFFFである場合、そのマークにはサムネール画像が付加されていない事を示す。

【0215】mark_nameは、マークの名前を示す。このフィールドの中の左からmark_name_lengthで示されるバイト数が、有効なキャラクター文字であり、前記名前を示す。このキャラクター文字は、UIAppInfoPlayListの中でcharacter_setによって示される方法で符号化されている。mark_nameフィールドの中で、それら有効なキャラクター文字に続くバイトの値は、どんな値が入っていても良い。

【0216】次に、"info.dvr"ディレクトリについて説明する。図38は、"info.dvr"ファイルのシンタクスを示す図である。"info.dvr"ディレクトリは、UIAppInfoVolume(), TableOfPlayLists()を持つ。TableOfPlayLists_Start_addressは、info.dvrファイルの先頭のバイトからの相対バイト数を単位として、TableOfPlayLists()の先頭アドレスを示す。相対バイト数はゼロからカウントされる。

【0217】図39に、UIAppInfoPlayList()のシンタクスを示す。ref_to_thumbnail_indexは、マークに付加されるサムネール画像の情報を示す。ref_to_thumbnail_indexフィールドが、0xFFFFでない値の場合、そのマークにはサムネール画像が付加されており、そのサムネール画像は、menu.tdatファイルの中にストアされている。その画像は、menu.tidxファイルの中でthumbnail_indexの値を用いて参照される（後述）。ref_to_thumbnail_indexフィールドが、0xFFFFである場合、そのマークにはサムネール画像が付加されていない事を示す。

【0218】その他のシンタクスフィールドは、Volumeについてのユーザインターフェースアプリケーションのパラメータを示すものであるが、本発明の実施の形態を説明するために必要ないので、詳細な説明を省略する。

【0219】図40は、TableOfPlayLists()のシンタクスを示す。TableOfPlayLists()は、PlayList(Real PlayListとVirtual PlayList)のファイル名をストアする。ボリューム（ディスク）に記録されているすべてのPlayListファイルは、TableOfPlayList()の中に含まれる。TableOfPlayLists()は、ボリュームの中のPlayListのデフォルトの再生順序を示す。lengthは、このlengthフィールドの直後からTableOfPlayLists()の最後までのTableOfPlayLists()のバイト数を示す。number_of_PlayListsは、ボリュームに記録されているPlayListの数を示す。PlayList_file_nameの10バイトの数字は、PlayListのファイル名を示す。

【0220】次に、サムネールについての情報をストアするファイルの内容を説明する。"menu.tidx"と"menu.tdat"は、メニュー・サムネール、すなわちVolumeを代表す

る1つのピクチャおよびPlayList毎にそれを代表する1つのピクチャの情報をストアする。すべてのメニュー・サムネールのヘッダ情報は、1つのmenu. tidxに集めて管理される。すべてのメニュー・サムネールのピクチャデータは、1つのmenu. tdatに集めて管理される。

【0221】"mark. tidx"と"mark. tdat"は、マーク・サムネール、すなわちマーク点で指されるピクチャについての情報をストアする。Volume中のすべてのClipおよびPlayListに付加されているすべてのマーク・サムネールのヘッダ情報は、1つのmark. tidxに集めて管理される。すべてのマーク・サムネールのピクチャデータは、1つのmark. tdatに集めて管理される。サムネールのピクチャデータは、例えば、画像をJPEGで符号化したデータである。

【0222】これら4個のファイルのシンタクスとセマンティクスを説明する。"menu. tidx"と"mark. tidx"は、同じシンタクス構造を持つ。図41は、"menu. tidx"と"mark. tidx"のシンタクス構造を示す。version_numberは、このサムネールヘッダ情報ファイルのバージョンナンバーを示す4個の数字である。lengthは、このlengthフィールドの直後のバイトからmenu. tidx/mark. tidxの最後のバイトまでのバイト数である。number_of_thumbnailsは、menu. tidxの場合にはmenu. tdatにストアされているサムネールピクチャの数であり、mark. tidxの場合にはmark. tdatにストアされているサムネールピクチャの数である。

【0223】tn_block_sizeは、menu. tidxの場合にはmenu. tdatの中の1つのtn_blockのサイズを示し、mark. tidxの場合にはmark. tdatの中の1つのtn_blockのサイズを示す。このサイズは、1024バイトを単位とする大きさである。例えば、tn_block_size=1は、1つのtn_blockのサイズが1024バイトであることを示す。1つのサムネールピクチャは、1つのtn_blockの中にストアされなければならない。number_of_tn_blocksは、menu. tidxの場合にはmenu. tdatの中にあるtn_blockの数を示し、mark. tidxの場合にはmark. tdatの中にあるtn_blockの数を示す。thumbnail_indexは、このthumbnail_indexフィールドに続くサムネール情報のインデックス番号を表す。thumbnail_indexして、0xFFFFという値を使用してはならない。

【0224】menu. tidxの場合、thumbnail_indexはUIAppInfoVolume(), UIAppInfoPlayList()の中のref_to_thumbnail_indexによって参照される。mark. tidxの場合、thumbnail_indexはPlayListMark()およびClipMark()の中のref_to_thumbnail_indexによって参照される。ref_to_tn_block_idは、menu. tidxの場合にはmenu. tdat中の1つのtn_blockを示し、そのtn_blockは、thumbnail_indexで指されるピクチャデータをストアしている。ref_to_tn_block_idの値は、menu. tdatのシンタクス中のtn_block_idの値を参照する。mark. tidxの場合にはmark. tda

t中の1つのtn_blockを示し、そのtn_blockは、thumbnail_indexで指されるピクチャデータをストアしている。ref_to_tn_block_idの値は、menu. tdatのシンタクス中のtn_block_idの値を参照する。

【0225】picture_byte_sizeは、thumbnail_indexで指される1つの符号化サムネールピクチャのデータ長をバイト単位で示す。picture_byte_sizeは、1024*tn_block_sizeの値以下でなければならない。すなわち、記録機は1つの符号化サムネールピクチャのデータ長を1024*tn_block_sizeの値以下になるように、符号化しなければならない。horizontal_picture_sizeは、thumbnail_indexで指される符号化サムネールピクチャの水平方向の画素数を示す。vertical_picture_sizeは、thumbnail_indexで指される符号化サムネールピクチャの垂直方向の画素数を示す。display_aspect_ratioは、thumbnail_indexで指される符号化サムネールピクチャのディスプレイ・アスペクト・レシオを示す。

【0226】図42は、"menu. tdat"と"mark. tdat"のシンタクス構造を示す。"menu. tdat"と"mark. tdat"は、同じシンタクス構造を持つ。tn_blockは、1つの符号化サムネールピクチャがストアされる領域である。一つのサムネールピクチャのバイト長は、1つのtn_blockの大きさ以下である。1つのピクチャデータの第一バイト目は、tn_blockの第一バイト目と一致していかなければならない。

【0227】menu. tdatの場合、1つのtn_blockのサイズはmenu. tdatの中のtn_block_sizeで示される。mark. tdatの場合、1つのtn_blockのサイズはmark. tdatの中のtn_block_sizeで示される。各tn_blockは、それがシンタクス中のfor-loopの中で現れるときのtn_block_idの値で区別される。menu. tidx中のtn_block_idは、menu. tidx中のref_to_tn_block_idによって参照される。mark. tidx中のtn_block_idは、mark. tidx中のref_to_tn_block_idによって参照される。サムネールは頻繁に追加、削除されるので、追加操作と部分削除の操作は容易に高速に実行できなければならない。この理由のため、menu. tdatとmark. tdatはブロック構造を有する。一つのピクチャデータは1つのtn_blockに格納される。

【0228】menu. tdatおよびmark. tdatのtn_block列の中に、使用されていないtn_blockが存在してもよい。例えば、あるサムネールの削除をする場合、サムネールのヘッダ情報ファイルの中にエントリーされているthumbnail_indexを消去し、サムネールのピクチャデータファイルを何も変更しなかったとき、tn_block列の中に、使用されていないtn_blockができる。

【0229】図43は、サムネールピクチャデータがどのようにtn_blockに格納されるかを模式的に表した図である。図のように、一つのサムネールピクチャのバイト長は、1つのtn_blockの大きさ以下である。tn_block列の中に、使用されていないtn_blockが存在してもよい。

【0230】次に、PlayListの編集動作のコンセプトについて説明する。以下の処理は、ユーザからの操作に基づいて、例えば、後述する図4-4の制御部17により実行される。

【0231】図4-5は、AVストリームが新しいClipとして記録される時のClipとPlayListの関係のコンセプトを説明する図である。AVストリームが新しいClipとして記録される場合、そのClip全体の再生可能範囲を参照するReal PlayListが作られる。

【0232】図4-6は、Virtual PlayListの作成のコンセプトについて説明する図である。ユーザがReal PlayListの再生範囲の中から、IN-timeとOUT-timeを指定することによって、見たい再生区間のPlayItemを作り、Virtual PlayListをつくる。

【0233】図4-7は、Real PlayListの再生区間の一部分を消去したときのClipとPlayListの関係のコンセプトを説明する図である。必要なClip AVストリームの再生部分だけを参照するように、Real PlayListのPlayItemを変更する。そして、Clip AVストリームの不必要的ストリーム部分を消去する。図4-7に示すように、Clip AVストリームの中央部のデータを消去しても、Clip AVストリームファイルは分割されないで、1つのファイルである。1つのClip AVストリームのデータを部分的に消去しても、残ったデータ部分は1つのClip AVストリームにまとめられる。

【0234】Real PlayListが変更されて、それが参照するClipのストリーム部分が消去された時、それと同じClipを使用しているVirtual PlayListが参照するClipがなくなつて、問題が起きるかもしれない。そのようなことがないように、ユーザインターフェースは、次に示す対策をとるべきである。

【0235】その“消去”の操作に対して、ユーザに「そのReal PlayListが参照しているClipのストリーム部分を参照しているVirtual PlayListが存在し、もし、そのReal PlayListが消去されると、そのVirtual PlayListもまた消去されることになるが、それで良いか？」と確認、警告する。または、前記Virtual PlayListを消去する代わりに、Real PlayListに対して次に示す“ミニマイズ(Minimize)”の操作をする。

【0236】図4-8は、ミニマイズの編集をしたときのClipとReal PlayList、Virtual PlayListの関係のコンセプトを説明する図である。ミニマイズ編集は、Real PlayListのPlayItemを、Virtual PlayListに必要なClipのストリーム部分だけを参照するように変更する。そして、Virtual PlayListにとって不必要的Clipのストリーム部分を消去する。

【0237】図4-8に示すように、Clip AVストリームの中央部のデータを消去しても、Clip AVストリームファイルは分割されないで、1つのファイルである。1つのClip AVストリームのデータを部分的に消去しても、残

ったデータ部分は1つのClip AVストリームにまとめられる。

【0238】次に、前記のコンセプトに基づいて、Clip AVストリームのデータを部分的に消去する場合のClip Information fileの変化について説明する。

【0239】前述したように、AVストリームをClipファイルとして新たに記録する時、そのClipはATCの不連続点を含まず、ただ1つのATC-sequenceを持つ。そして、ATCの不連続点は、編集等によってClip AVストリームファイルのストリームデータを部分的に消去した場合にだけ、作られることを想定している。すなわち、図4-7や図4-8に示すように、1つのClip AVストリームのデータを部分的に消去して、残ったデータ部分が1つのClip AVストリームにまとめられた時、そのClipはATCの不連続点を持ち、複数のATC-sequenceを持つ。例えば、図4-9において、編集前のClipはATCの不連続点を含まず、ただ1つのATC-sequenceを持つとする。そして図に示すように、Clip AVストリームの中央部のデータを消去した場合、編集後のClipは、2個のATC-sequenceを持つ。

【0240】図5-0は、1つのClip AVストリームのデータを部分的に消去した時のATC-sequence、STC-sequenceおよびprogram-sequenceの関係を説明する図である。編集前のClipは、ただ1つのATC-sequenceと1つのSTC-sequenceと1つのprogram-sequenceを持つとする。すなわち、このClipの中では、program-sequenceの内容が変化しないとする。この時、図に影で示す部分のAVストリームデータを消去したとする。編集の結果、Clipは3個のATC-sequenceと3個のSTC-sequenceを持つ、一方、program-sequenceは1個のままである。このprogram-sequenceは、ATC-sequenceの境界とSTC-sequenceの境界をまたいでいる。

【0241】次に、前記のようにClip AVストリームのデータが部分的に消去される時の、ClipとPlayListの関係について説明する。

【0242】図5-1は、CPIがEP_mapであるClip AVストリームの一部分を消去した時のClipとPlayListの関係を説明する図である。編集前のClipは1個のATC-sequenceと3個のSTC-sequenceを持つとする。このATC-sequenceについてのoffset_STC_id[0]はゼロである。そして、Clipの中のstc=1であるSTC-sequenceは、PlayItem2とPlayItem3に使われているとする。今、図に示すようにstc=1であるSTC-sequenceのAVストリームデータについて、PlayItem2とPlayItem3に使われていない部分のAVストリームデータを消去したとする。

【0243】編集後のClipは2個のATC-sequenceを持ち、stc_id=1であったSTC-sequenceは2個のSTC-sequenceに分かれる。1番目のATC-sequenceについてのoffset_STC_id[0]はゼロにセットされ、2番目のATC-sequenceについてのoffset_STC_id[1]は1にセットされる。すなわち、1番目のATC-sequenceの中で最後のSTC-sequence

のstc_id と 2番目のATC-sequenceの中で最初のSTC-sequenceのstc_id は、同じ値で1になる。

【0244】これにより、編集後のVirtual PlayListのPlayItem3とPlayItem4のref_to_STC_idの値を変更する必要がない。Clip AVストリームファイルの部分的なデータを消去する時に、その消去部分を使用していないVirtual PlayListについては何も変更しなくても良い。

【0245】このように、Clip AVストリームの中にATCの不連続点を作ることができるので、Clip AVストリームのミドル(middle)部分のストリームデータを消去した場合に、Clipファイルを2個に分割する必要がない。さらに、ATCシーケンス毎に、その上にある最初のSTCシーケンスのSTC_idに対するoffset_STC_idを用いることにより、Clip AVストリームファイルの部分的なデータを消去する時に、その消去部分を使用していないVirtual PlayListについては何も変更しなくて良い。

【0246】この効果についての理解を助けるために、図52はClipの中にATCの不連続を許さない場合において、Clip AVストリームの一部分を消去した時に、Clipファイルが二つに分かれる場合を説明する図であり、また、その時のClipとPlayListの関係を説明する図である。

【0247】図51と同様にして、編集前のClipは1個のATC-sequenceと3個のSTC-sequenceを持つとする。このATC-sequenceについてのoffset_STC_id[0]はゼロである。そして、Clipの中のstc=1であるSTC-sequenceは、PlayItem2とPlayItem3に使われているとする。今、図に示すようにstc=1であるSTC-sequenceのAVストリームデータについて、PlayItem2とPlayItem3に使われていない部分のAVストリームデータを消去したとする。

【0248】Clipの中にATCの不連続を許さない場合、編集後はClip-AとClip-Bの2個のClipファイルに分かれる。そのため、PlayItem3とPlayItem4が参照するClipファイルの名前を変更する必要がある。すなわち、Clip AVストリームファイルの部分的なデータを消去する時に、その消去部分を使用していないVirtual PlayListであっても、その内容を変更しなければならない場合がある。

【0249】Clipの中にATCの不連続を許さない場合は、これを許す場合に比べると次の問題がある。

(1) ディスク中のClipファイル数が多くなる問題。これによって、ディスクの再生開始時に全てのClipファイルを読み出し、再生装置(動画像記録再生装置1)のメモリ(制御部17に内蔵されているメモリ)にストアする処理にかかる時間が増える問題がある。また、ディスク(記録媒体10)中に記録可能なファイル数の上限がある所定の値に決めた時、編集等によりClipファイル数が多くなり、その数が上限に達してしまい、ディスク中にまだ空き領域があるのに記録できなくなる問題が発生する。

(2) Clip AVストリームファイルの部分的なデータを消去した時に、ディスク中にあるVirtual PlayListの変更にかかる時間が大きい問題。

【0250】本発明は、これらの問題を解決する。すなわち、ディスクの再生開始時に全てのClipファイルを読み出し、再生装置のメモリにストアする処理にかかる時間を小さくできる。また、ディスク中に記録可能なファイル数の上限をより小さく設定できる。また、Clip AVストリームファイルの部分的なデータを消去した時に、ディスク中にあるVirtual PlayListの変更にかかる時間が小さくなる。

【0251】図53は、CPIがTU_mapであるClip AVストリームの一部分を消去した時のClipとPlayListの関係を説明する図である。編集前のClipは1個のATC-sequenceを持つ。このATC-sequenceについてのoffset_arrival_time[0]はゼロである。Virtual PlayListのPlayItem1, PlayItem2, PlayItem3およびPlayItem4は、このATC-sequenceを参照しているとする。今、図に示すようにATC-sequenceのAVストリームデータについて、どのPlayItemにも使われていないAVストリームデータを消去したとする。

【0252】編集後のClipは2個のATC-sequenceを持つ。1番目のATC-sequenceについてのoffset_arrival_time[0]はゼロにセットされ、2番目のATC-sequenceについてのoffset_arrival_time[1]は値Xにセットされる。値Xは、OUT_time2よりも大きく、IN_time3よりも小さい。すなわち、編集後にVirtual PlayListのPlayItem3とPlayItem4のIN_timeとOUT_timeの値を変更する必要がない。

【0253】Clip AVストリームファイルの部分的なデータを消去する時に、その消去部分を使用していないVirtual PlayListについては何も変更しなくても良い。

【0254】TU_map typeのPlayListを再生する場合、再生機はPlayItemのIN_timeとATC-sequenceのoffset_arrival_timeの値を比較することにより、そのIN_timeとOUT_timeが指すところのATC-sequenceを見つけることができる。例えば、図53の場合、PlayItem3のIN_time3は2番目のATC-sequenceのoffset_arrival_time(=X)よりも大きいので、PlayItem3のIN_time3とOUT_time3は、2番目のATC-sequenceを指すことがわかる。

【0255】次に、DVRアプリケーション構造のデータを記録再生するシステムについて、図44の動画像記録再生装置1のブロック図を用いて説明する。

【0256】例えば、光ディスクにより構成される記録媒体10は、再生部61の読み出し部11により、そこに記録されている情報が読み出される。復調部12は、読み出し部11が記録媒体10から読み出したデータを復調し、ECC復号部13に供給する。ECC復号部13は、復調部12より供給されたデータを、AVストリームとデータベースとに分離し、AVストリームをソースデパケッ

タイザ14に供給し、データベースを制御部17に出力する。

【0257】ソースデパケッタイザ14は、入力されたAVストリームをデパッケタイズし、デマルチプレクサ15に出力する。デマルチプレクサ15は、ソースデパケッタイザ14より供給されたデータをビデオ(V)、オーディオ(A)、およびシステム(S)の各データに分離し、AVデコーダ16とマルチプレクサ25に出力する。

【0258】AVデコーダ16は、入力されたビデオデータとオーディオデータを、システムデータに基づいてデコードし、ビデオ信号を端子18から、オーディオ信号を端子19から、それぞれ出力する。

【0259】記録部62のAVエンコーダ23には、端子21から入力されたビデオ信号と、端子22から入力されたオーディオ信号が供給される。ビデオ信号はまた、ビデオ解析部24にも供給される。AVエンコーダ23とビデオ解析部24には、端子21から入力されたビデオ信号の代わりに、必要に応じて、AVデコーダ16が出力したビデオ信号が供給される。

【0260】AVエンコーダ23は、入力されたビデオ信号とオーディオ信号をエンコードし、エンコードしたビデオ信号(V)、オーディオ信号(A)、およびエンコードに対応するシステムデータ(S)を、マルチプレクサ25に出力する。

【0261】ビデオ解析部24は、入力されたビデオ信号を解析し、解析結果を制御部17に出力する。

【0262】端子33には、デジタルインターフェースまたはデジタルテレビチューナからのトランスポートストリームが入力され、スイッチ27を介して、デマルチプレクサ15、またはさらにスイッチ28を介して、多重化ストリーム解析部26、およびソースパケッタイザ29に供給される。多重化ストリーム解析部26とソースパケッタイザ29にはまた、スイッチ28を介してマルチプレクサ25が出力した信号も、スイッチ27からの信号に代えて供給可能とされている。

【0263】多重化ストリーム解析部26は、入力された信号を解析し、解析結果を制御部17に出力する。ソースパケッタイザ29は、入力された信号をパケッタイズし、ECC符号化部30に供給する。ECC符号化部30には、制御部17が出力するデータベースも供給されている。

【0264】ECC符号化部30は、入力に誤り訂正符号を付加し、符号化し、変調部31に出力する。変調部31は、ECC符号化部30から入力されたデータを変調し、書き込み部32に出力する。書き込み部32は、変調部31から入力されたデータを記録媒体10に書き込む処理を実行する。

【0265】制御部17は、各種のデータを記憶する記憶部17Aを有しており、上述したフォーマットを管理

し、データの記録媒体10に対する記録または再生のために、各部を制御する。

【0266】制御部17にはまた、ドライブ41が接続されており、磁気ディスク51、光ディスク52、光磁気ディスク53、または半導体メモリ54などがドライブされる。

【0267】なお、光ディスク52は、記録媒体10と兼用することも可能である。

【0268】次に記録時の基本的動作について、動画像記録再生装置1自身が、入力オーディオビデオ信号を符号化して記録する場合を例として説明する。

【0269】記録部62の端子21と端子22からは、それぞれビデオ信号とオーディオ信号が入力される。ビデオ信号は、ビデオ解析部24とAVエンコーダ23へ入力される。また、オーディオ信号もまたAVエンコーダ23へ入力される。AVエンコーダ23は、入力ビデオ信号とオーディオ信号を符号化し、符号化ビデオストリーム(V)、符号化オーディオストリーム(A)、およびシステム情報(S)をマルチプレクサ25に出力する。

【0270】符号化ビデオストリーム(V)は、例えばMPEG2ビデオストリームであり、符号化オーディオストリーム(A)は、例えばMPEG1オーディオストリームやドルビーAC3(商標)オーディオストリーム等である。システム情報(S)は、ビデオオーディオの符号化情報(符号化ピクチャやオーディオフレームのバイトサイズ、ピクチャ符号化タイプ等)やAV同期等の時間情報である。

【0271】マルチプレクサ25は、入力ストリームを入力システム情報に基づいて多重化して、多重化ストリームを出力する。多重化ストリームは、例えば、MPEG2トランスポートストリームやMPEG2プログラムストリームである。多重化ストリームは、多重化ストリーム解析部26およびソースパケッタイザ29に入力される。ソースパケッタイザ29は、入力多重化ストリームを、記録媒体10のアプリケーションフォーマットに従って、ソースパケットから構成されるAVストリームに符号化する。AVストリームは、ECC(誤り訂正)符号化部30で誤り訂正符号が付加され、変調部31で変調処理されて、書き込み部32へ入力される。書き込み部32は、制御部17から指示される制御信号に基づいて、記録媒体10へAVストリームファイルを記録する。

【0272】次に、例えば、図示せぬデジタルインターフェースまたはデジタルTVチューナから入力されるデジタルTV放送等のトランスポートストリームを記録する場合を説明する。

【0273】端子33からはトランスポートストリームが入力される。入力トランスポートストリームの記録方法は、2通りあり、それらは、トランスペアレントに記録する方法と記録ビットレートを下げるなどの目的のために再エンコードをして記録する方法である。記録方法の指示情報は、ユーザインターフェースとしての端子2

0から制御部17へ入力され、制御部17が記録方法を制御する。

【0274】入力トランSPORTストリームをトランスペアレントに記録する場合、トランSPORTストリームは、多重化ストリーム解析部26およびソースパケットライザ29に入力される。これ以後、記録媒体10へAVストリームが記録されるまでの処理は、上述の入力オーディオ信号とビデオ信号を符号化して記録する場合と同じである。

【0275】入力トランSPORTストリームを再エンコードして記録する場合、入力トランSPORTストリームは、デマルチプレクサ15へ入力される。デマルチプレクサ15は、ビデオストリーム(V)をAVデコーダ16へ入力する。AVデコーダ16は、ビデオストリームを復号し、再生ビデオ信号をAVエンコーダ23へ入力する。AVエンコーダ23は、入力ビデオを符号化し、符号化ビデオストリーム(V)をマルチプレクサ25へ入力する。

【0276】一方、デマルチプレクサ15から出力されるオーディオストリーム(A)とシステム情報(S)は、ダイレクトにマルチプレクサ25へ入力される。マルチプレクサ25は、入力ストリームを、入力システム情報に基づいて多重化して、多重化ストリームを出力する。これ以後、記録媒体10へAVストリームが記録されるまでの処理は、上述の入力オーディオビデオ信号を符号化して記録する場合と同じである。

【0277】この動画像記録再生装置1は、AVストリームファイルを記録すると共に、そのファイルに関するアプリケーションデータベース情報もまた記録する。アプリケーションデータベース情報は、制御部17により作成される。制御部17への入力情報は、ビデオ解析部24からの動画像の特徴情報、多重化ストリーム解析部26からのAVストリームの特徴情報、およびユーザインタフェースとしての端子20から入力されるユーザの指示情報である。

【0278】ビデオ解析部24からの動画像の特徴情報は、動画像記録再生装置1自身がビデオ信号を符号化する場合において、動画像記録再生装置1により生成されるものである。ビデオ解析部24は、入力ビデオ信号の内容を解析し、入力動画像信号の中の特徴的なマーク点の画像に関する情報を生成する。この情報は、例えば、入力ビデオ信号の中のプログラムの開始点、シーンチェンジ点、CMのスタート・エンド点などの特徴的なマーク点の画像の指示情報であり、また、これには、その画像のサムネールも含まれる。これらの画像の指示情報は、制御部17を介して、マルチプレクサ25へ入力される。

【0279】マルチプレクサ25は、制御部17から指示されるマーク点の画像の符号化ピクチャを多重化した時に、その符号化ピクチャのAVストリーム上でのアドレス情報を制御部17に返す。制御部17は、特徴的な画

像の種類と、その符号化ピクチャのAVストリーム上でのアドレス情報を関連付けて記憶する。

【0280】多重化ストリーム解析部26からのAVストリームの特徴情報は、記録されるAVストリームの符号化情報に関する情報であり、これらは動画像記録再生装置1により生成される。例えば、AVストリームの中におけるIピクチャのタイムスタンプとアドレス情報、STCの不連続情報、プログラム内容の変化情報、アライバルタイムとアドレス情報、などが含まれる。

【0281】AVストリーム内のIピクチャのタイムスタンプとアドレス情報は、上述のEP_mapにストアされるデータとなる。AVストリーム内のSTCの不連続情報は、上述のSequenceInfoにストアされるデータとなる。AVストリーム内のプログラム内容の変化情報は、ProgramInfoにストアされるデータとなる。また、AVストリーム内のアライバルタイムとアドレス情報は、上述のTU_mapにストアされる。

【0282】また、多重化ストリーム解析部26は、端子33から入力されるトランSPORTストリームをトランスペアレントに記録する場合、AVストリームの中の特徴的なマーク点の画像を検出し、その種類とアドレス情報を生成する。この情報は、ClipMarkにストアされるデータとなる。

【0283】多重化ストリーム解析部26からのAVストリームの特徴情報は、AVストリームのデータベース(ClipInformation)にストアされるものである。

【0284】端子20からのユーザの指示情報には、AVストリームの中のお好みの再生区間の指定情報、その再生区間の内容を説明するキャラクター文字、ユーザがお好みのシーンにセットするブックマークやリジューム点のAVストリームの中のタイムスタンプなどが含まれる。これらのユーザの指示情報は、PlayListのデータベースにストアされるものである。

【0285】制御部17は、前記入力情報に基づいて、AVストリームのデータベース(ClipInformation)、PlayListのデータベース、記録媒体10の記録内容の管理情報(info.dvr)、およびサムネール情報を生成する。これらのデータベース情報は、AVストリームと同様にして、ECC(誤り訂正)符号化部30、変調部31で処理されて、書き込み部32へ入力される。書き込み部32は、制御部17から指示される制御信号に基づいて、このデータベース情報を、記録媒体10へ、アプリケーションデータベース情報として記録する。

【0286】次に、再生時の基本的な動作について説明する。

【0287】記録媒体10には、AVストリームファイルとアプリケーションデータベース情報が記録されている。

【0288】はじめに制御部17は、再生部61の読み出し部11に対して、アプリケーションデータベース情

報を読み出すように指示する。そして、読み出し部11は、記録媒体10からアプリケーションデータベース情報を読み出し、そのデータベース情報は、復調部12、ECC(誤り訂正)復号部13の処理を経て、制御部17へ入力される。

【0289】制御部17は、アプリケーションデータベースに基づいて、記録媒体10に記録されているPlayListの一覧を、端子20のユーザインターフェースへ出力する。ユーザは、PlayListの一覧から再生したいPlayListを選択し、再生を指定されたPlayListが制御部17へ入力される。制御部17は、そのPlayListの再生に必要なAVストリームファイルの読み出しを読み出し部11に指示する。そして、読み出し部11は、記録媒体10からそのAVストリームを読み出し、AVストリームは復調部12、ECC復号部13の処理を経て、ソース・デパッケッタイザ14へ入力される。

【0290】ソース・デパッケッタイザ14は、記録媒体のアプリケーションフォーマットのAVストリームを、デマルチプレクサ15へ入力できるストリームに変換する。デマルチプレクサ15は、制御部17により指定されたAVストリームの再生区間(PlayItem)を構成するビデオストリーム(V)、オーディオストリーム(A)、およびシステム情報(S)をAVデコーダ16へ入力する。AVデコーダ16は、ビデオストリームとオーディオストリームを復号し、再生ビデオ信号と再生オーディオ信号を、それぞれ端子18と端子19から出力する。

【0291】ユーザによって選択されたEP_mapタイプのPlayListをある時間から途中再生する場合、制御部17は、指定された時間にもっとも近いPTSを持つIピクチャのアドレスからデータを読み出すように読み出し部11へ指示する。

【0292】また、ユーザによって選択されたTU_mapタイプのPlayListをある時間から途中再生する場合、制御部17は、指定された時間にもっとも近いアライバルタイムのソースパケットのアドレスからデータを読み出すように読み出し部11へ指示する。

【0293】さらに、Clip Informationの中のClipMarkにストアされている番組の頭出し点やシーンチェンジ点の中から、ユーザがあるマークを選択した時(例えば、この選択動作は、ClipMarkにストアされている番組の頭出し点やシーンチェンジ点のサムネール画像リストをユーザインターフェースに表示して、ユーザが、その中からある画像を選択することにより行われる)、制御部17は、Clip Informationの内容に基づいて、記録媒体10からのAVストリームの読み出し位置を決定し、そのAVストリームの読み出しを読み出し部11へ指示する。

【0294】すなわち、ユーザが選択した画像がストアされているAVストリーム上でのアドレスに最も近いアドレスにあるIピクチャからのデータを読み出すように読み出し部11へ指示が出される。読み出し部11は、指

定されたアドレスからデータを読み出し、読み出されたデータは、復調部12、ECC復号部13の処理を経て、デマルチプレクサ15へ入力され、AVデコーダ16で復号されて、マーク点のピクチャのアドレスで示されるAVデータが再生される。

【0295】次に、ユーザが、AVストリームの編集をする場合を説明する。

【0296】ユーザが、記録媒体10に記録されているAVストリームの再生区間を指定して新しい再生経路を作成したい場合、ユーザインターフェースの端子20から、再生区間のイン点とアウト点の情報が制御部17へ入力される。制御部17は、AVストリームの再生区間(PlayItem)をグループ化したもの(PlayList)のデータベースを作成する。

【0297】ユーザが、記録媒体10に記録されているAVストリームの一部を消去したい場合、ユーザインターフェースの端子20から、消去区間の情報が制御部17へ入力される。制御部17は、必要なAVストリーム部分だけを参照するようにPlayListのデータベースを変更する。また、AVストリームの不必要的ストリーム部分を消去するように、書き込み部32に指示する。また、Clip AVストリームの変化に基づいて、そのClip Information fileの内容を変更する。

【0298】ユーザが、記録媒体10に記録されているAVストリームの再生区間を指定して新しい再生経路を作成したい場合であり、かつそれぞれの再生区間をシームレスに接続したい場合の動作を説明する。この場合、制御部17は、AVストリームの再生区間(PlayItem)をグループ化したもの(PlayList)のデータベースを作成し、さらに、再生区間の接続点付近のビデオストリームの部分的な再エンコードと再多重化が必要になる。

【0299】まず、ユーザインターフェースとしての端子20から、再生区間のイン点のピクチャとアウト点のピクチャの情報が制御部17へ入力される。制御部17は、読み出し部11に、イン点のピクチャとアウト点のピクチャを再生するために必要なデータの読み出しを指示する。そして、読み出し部11は、記録媒体10からそのデータを読み出し、そのデータは、復調部12、ECC復号部13、ソース・デパッケッタイザ14を経て、デマルチプレクサ15へ入力される。

【0300】制御部17は、デマルチプレクサ15へ入力されたストリームを解析して、ビデオストリームの再エンコード方法(picture_coding_typeの変更、再エンコードする符号化ビット量の割り当て)と再多重化方法を決定して、その方法をAVエンコーダ23とマルチプレクサ25へ供給する。

【0301】次に、デマルチプレクサ15は、入力されたストリームをビデオストリーム(V)、オーディオストリーム(A)、およびシステム情報(S)に分離する。ビデオストリームは、「AVデコーダ16へ入力されるデータ」

と、「マルチプレクサ25へ直接入力されるデータ」がある。前者のデータは、再エンコードするために必要なデータであり、これはAVデコーダ16で復号され、復号されたピクチャは、AVエンコーダ23で再エンコードされて、ビデオストリームになる。後者のデータは、再エンコードをしないで、オリジナルのストリームからコピーされるデータである。オーディオストリームとシステム情報は、マルチプレクサ25に直接入力される。

【0302】マルチプレクサ25は、制御部17から入力された情報に基づいて、入力ストリームを多重化し、多重化ストリームを出力する。多重化ストリームは、EC C(誤り訂正)符号化部30、変調部31で処理され、書き込み部32へ入力される。書き込み部32は、制御部17から指示される制御信号に基づいて、記録媒体10へAVストリームを記録する。

【0303】次に、図54は、AVストリームをClipをして新しく記録するときの、Clip AVストリームファイルと、それに関連するClip Informationファイルの、動画像記録再生装置1の記録動作のフローチャートを示す。

【0304】ステップS11で、制御部17は、端子21および22から入力されるAV入力をエンコードして得たトランスポートストリーム、または端子33のデジタルインターフェースから入力されるトランスポートストリームをファイル化して、Clip AVストリームファイルを作成して記録する。

【0305】ステップS12で、制御部17は上記AVストリームファイルについてのClipInfo(図8)を作成する。

【0306】ステップS13で、制御部17は上記AVストリームファイルについてのSequenceInfo(図13)を作成する。

【0307】ステップS14で、制御部17は上記AVストリームファイルについてのProgramInfo(図15)を作成する。

【0308】ステップS15で、制御部17は上記AVストリームファイルについてのCPI(EP-mapまたはTU-map)(図24、図25および図26)を作成する。

【0309】ステップS16で、制御部17は上記AVストリームファイルについてのClipMarkを作成する。

【0310】ステップS17で、制御部17は上記ClipInfo, SequenceInfo, ProgramInfo, CPI, およびClipMarkがストアされたClip Informationファイル(図8)を記録する。

【0311】なお、ここでは各処理を時系列に説明したが、ステップS11からステップS116は、実際には同時に動作するものである。

【0312】次に、AVストリームをClipをして新しく記録するときの、SequenceInfo(図13)の作成の動作例を、図55のフローチャートを用いて説明する。この処理は、図44の多重化ストリーム解析部26で行われ

る。

【0313】ステップS31において、制御部17は最初のトランスポートパケットをATCシーケンスの開始点とする。すなわち、SPN_ATC_startが設定される。また、このとき、atc_idとstc_idも設定される。

【0314】ステップS32において、多重化ストリーム解析部26はAVストリームに含まれるアクセスユニット(例えば、ピクチャやオーディオフレーム)のPTSを解析する。

【0315】ステップS33において、多重化ストリーム解析部26はPCRパケットが受信されたかどうかを調べる。ステップS33において、Noの場合はステップS32へ戻り、Yesの場合はステップS34へ進む。

【0316】ステップS34において、多重化ストリーム解析部26はSTCの不連続が検出されたか否かを調べる。Noの場合は、ステップS32へ戻る。YESの場合は、ステップS35へ進む。なお、記録開始後、最初に受信されたPCRパケットの場合は、必ずステップS35へ進む。

【0317】ステップS35において、多重化ストリーム解析部26は、新しいSTCの最初のPCRを伝送するトランスポートパケットの番号(アドレス)を取得する。

【0318】ステップS36において、制御部17は上記パケット番号をSTCシーケンスの開始するソースパケット番号として取得する。すなわち、SPN_STC_startが設定される。また、このとき、新たなstc_idが設定される。

【0319】ステップS37において、制御部17はSTCシーケンスの表示開始のPTSと表示終了のPTSを取得し、それぞれ、presentation_start_time、またはpresentation_end_timeに設定し、それらに基づいて、SequenceInfo(図13)を作成する。

【0320】ステップS38において、制御部17は最後のトランスポートパケットが入力終了したかどうかを調べる。Noの場合は、ステップS32へ戻り、Yesの場合は処理を終了する。

【0321】ProgramInfo(図15)の作成の動作例を図56のフローチャートを用いて説明する。この処理は図44の多重化ストリーム解析部26で行われる。

【0322】ステップS51において、多重化ストリーム解析部26はPSI/SIを含むトランスポートパケットが受信されたかどうかを調べる。ここで、PSI/SIのトランスポートパケットは、具体的には、PAT, PMT, SITのパケットである。SITは、DVB規格で規定されているパーシャルトランスポートストリームのサービス情報が記述されているトランスポートパケットである。ステップS51において、Noの場合はステップS51へ戻り、Yesの場合はステップS52へ進む。

【0323】ステップS52において、多重化ストリーム解析部26は、PSI/SIの内容が変わったかを調べる。

すなわち、PAT, PMT, SITのそれぞれの内容が、以前に受信したそれぞれの内容と比べて変化したかどうかが調べられる。内容が変化していない場合は、ステップS 5 1へ戻る。内容が変化した場合は、ステップS 5 3へ進む。なお、記録開始後、最初に受信されたPSI/SIの場合は、必ずステップS 5 3へ進む。

【0324】ステップS 5 3において、制御部17は新しいPSI/SIを伝送するトランSPORTパケットの番号(アドレス)とその内容を取得する。

【0325】ステップS 5 4において、制御部17はProgram-sequenceの情報を作成し、ProgramInfo(図10)を作成する。

【0326】ステップS 5 5において、制御部17は最後のトランSPORTパケットが入力終了したかどうかを調べる。Noの場合は、ステップS 5 1へ戻り、Yesの場合は処理を終了する。

【0327】次にEP_map(図24)の作成の動作例を、図57のフローチャートを用いて説明する。この処理は図44の多重化ストリーム解析部26で行われる。

【0328】ステップS 7 1で多重化ストリーム解析部26は、記録するAVプログラムのビデオのPIDをセットする。トランSPORTストリームの中に複数のビデオが含まれている場合は、それぞれのビデオPIDがセットされる。

【0329】ステップS 7 2で多重化ストリーム解析部26は、ビデオのトランSPORTパケットを受信する。

【0330】ステップS 7 3で多重化ストリーム解析部26は、トランSPORTパケットのペイロード(パケットヘッダーに続くデータ部)がPESパケットの第1バイト目から開始しているかを調べる(PESパケットは、MPE G2で規定されているパケットであり、エレメンタリーストリームをパケット化するものである)。これは、トランSPORTパケットヘッダにある"payload_unit_start_indicator"の値を調べることによりわかり、この値が1である場合、トランSPORTパケットのペイロードがPESパケットの第1バイト目から開始する。ステップS 7 3でNoの場合は、ステップS 7 2へ戻り、Yesの場合は、ステップS 7 4へ進む。

【0331】ステップS 7 4で多重化ストリーム解析部26は、PESパケットのペイロードが、MPEGビデオのsequence_header_code(32ビット長で"0x000001B3"の符号)の第1バイト目から開始しているかを調べる。ステップS 7 4でNoの場合は、ステップS 7 2へ戻り、Yesの場合は、ステップS 7 5へ進む。

【0332】ステップS 7 5へ進んだ場合、制御部17は現在のトランSPORTパケットをエントリーポイントとする。

【0333】ステップS 7 6で制御部17は、上記パケットのパケット番号と上記sequence_header_codeから開始するIピクチャのPTSとそのエントリーポイントが属

するビデオのPIDを取得し、EP_mapを作成する。

【0334】ステップS 7 7で、多重化ストリーム解析部26は、現在のパケットが最後に入力されるトランSPORTパケットであるかどうかを判定する。最後のパケットでない場合、ステップS 7 2へ戻る。最後のパケットである場合、処理を終了する。

【0335】図58は、Real PlayListの作成方法を説明するフローチャートを示す。図44の動画像記録再生装置1のブロック図を参照しながら説明する。

【0336】ステップS 9 1で、制御部17はClip AVストリームを記録する。

【0337】ステップS 9 2で、制御部17は上記Clipの全ての再生可能範囲をカバーするPlayItem(図36)からなるPlayList()(図31)を作成する。Clipの中にSTC不連続点があり、PlayList()が2つ以上のPlayItemからなる場合は、制御部17はPlayItem間のconnection_conditionもまた決定する。

【0338】ステップS 9 3で、制御部17はUIAppInfoPlayList()を作成する。UIAppInfoPlayList()はPlayListの内容をユーザーへ説明するための情報を含む。本実施の形態ではその説明を省略する。

【0339】ステップS 9 4で、制御部17はPlayListMarkを作成する(本実施の形態ではその説明を省略)。

【0340】ステップS 9 5で、制御部17はMakersPrivateDataを作成する(本実施の形態ではその説明を省略)。

【0341】ステップS 9 6で、制御部17はReal Playlistファイルを記録する。

【0342】このようにして、新規にClip AVストリームを記録する毎に、1つのReal Playlistファイルが作られる。

【0343】図59は、Virtual PlayListの作成方法を説明するフローチャートである。

【0344】ステップS 1 1 1で、ユーザーインターフェースを通して、ディスク(記録媒体10)に記録されている1つのReal PlayListの再生が指定される。そして、そのReal PlayListの再生範囲の中から、ユーザーインターフェースを通して、IN点とOUT点で示される再生区間が指定される。

【0345】ステップS 1 1 2で、制御部17はユーザーによる再生範囲の指定操作がすべて終了したか調べる。ユーザーが上記指示した再生区間に続けて再生する区間を選ぶ場合はステップS 1 1 1へ戻る。

【0346】ステップS 1 1 2でユーザーによる再生範囲の指定操作がすべて終了したと判定された場合は、ステップS 1 1 3へ進む。

【0347】ステップS 1 1 3で、連続して再生される2つの再生区間の間の接続状態(connection_condition)を、ユーザーがユーザーインターフェースを通して決定するか、または制御部17が決定する。

【0348】ステップS114で、ユーザーインターフェースを通して、ユーザーがサブパス(アフレコ用オーディオ)情報を指定する。ユーザーがサブパスを作成しない場合はこのステップの処理はスルーされる。サブパス情報は、PlayListの中のSubPlayItemにストアされる情報であるが、本発明の趣旨に必要ないので説明を省略する。

【0349】ステップS115で、制御部17はユーザーが指定した再生範囲情報、およびconnection_conditionに基づいて、PlayList() (図28) を作成する。

【0350】ステップS116で、制御部17はUIAppInfoPlayList()を作成する。UIAppInfoPlayList()はPlayListの内容をユーザーへ説明するための情報を含む。本実施の形態ではその説明を省略する。

【0351】ステップS117で、制御部17はPlayListMarkを作成する(本実施の形態ではその説明を省略)。

【0352】ステップS118で、制御部17はMakers PrivateDataを作成する(本実施の形態ではその説明を省略)。

【0353】ステップS119で、制御部17はVirtual PlayListファイルを記録媒体10に記録する。

【0354】このようにして、記録媒体10に記録されているReal PlayListの再生範囲の中から、ユーザーが見たい再生区間を選択してその再生区間をグループ化したもの毎に、1つのVirtual PlayListファイルが作られる。

【0355】図60はPlayListの再生方法を説明するフローチャートである。

【0356】ステップS131で、制御部17はInfo.dvr, Clip Information file, PlayList fileおよびサムネールファイルの情報を取得し、ディスク(記録媒体10)に記録されているPlayListの一覧を示すGUI画面を作成し、ユーザーインターフェースを通して、GUIに表示する。

【0357】ステップS132で、制御部17はそれぞれのPlayListのUIAppInfoPlayList()に基づいて、PlayListを説明する情報をGUI画面に提示する。

【0358】ステップS133で、ユーザーインターフェースを通して、GUI画面上からユーザーが1つのPlayListの再生を指示する。

【0359】ステップS134で、制御部17は現在のPlayItemのSTC-idとIN_timeのPTSから、IN_timeより時間的に前で最も近いエントリーポイントのあるソースパケット番号を取得する。

【0360】ステップS135で、制御部17は上記エントリーポイントのあるソースパケット番号からAVストリームのデータを読み出し、デコーダへ供給する。

【0361】ステップS136で、現在のPlayItemの時間的に前のPlayItemがあった場合は、制御部17は、前

のPlayItemと現在のPlayItemとの表示の接続処理をconnection_conditionに従って行う。

【0362】ステップS137で、制御部17は、AVデコーダ16にIN_timeのPTSのピクチャから表示を開始するように指示する。

【0363】ステップS138で、制御部17は、AVデコーダ16にAVストリームのデコードを続けるように指示する。

【0364】ステップS139で、制御部17は、現在表示の画像が、OUT_timeのPTSの画像か否かを調べる。Noの場合は、ステップS140へ進む。ステップS140で現在の画像を表示して、ステップS138へ戻る。ステップ38で現在表示の画像がOUT_timeのPTSの画像の場合はステップ40へ進む。

【0365】ステップS141で、制御部17は、現在のPlayItemがPlayListの中で最後のPlayItemかを調べる。Noの場合はステップS134へ戻る。Yesの場合は、PlayListの再生を終了する。

【0366】このようなシンタクス、データ構造、規則に基づく事により、記録媒体10に記録されているデータの内容、再生情報などを適切に管理することができ、もって、ユーザーが再生時に適切に記録媒体に記録されているデータの内容を確認したり、所望のデータを簡便に再生できるようにすることができる。

【0367】次に、本発明のAVストリームファイルとデータベースファイルが記録されている記録媒体から、デジタルバス経由で、別の記録媒体へAVストリームファイルとデータベースファイルをファイルコピーする方法を説明する。

【0368】まず、本方法の目的を説明するために、AVストリームとそのデータベースを共にデータ転送する場合と、AVストリームだけをデータ転送する場合の違いを説明する。

【0369】図61、図62は、AVストリームとそのデータベースを共にデータ転送する場合を示す。一方、図63、図64は、AVストリームだけをデータ転送する場合を示す。

【0370】コピー元の記録媒体(例えば、後述する図71の記録媒体10)には、ClipとPlayListが記録されている。説明を簡単化するために、いま、1個のClip(ClipInformation fileとClip AV stream file)を使用しているPlayListファイルがコピー元の記録媒体に記録されているとする。

【0371】図61は、当該PlayListとClipのファイルを、コピー元(図71の再生装置2)からコピー先(図71の記録装置3)へ、IEEE1394のデジタルバス(図71のデジタルバス60)を経由して、Asynchronous(アシンクロナス)転送によって、ファイル転送する場合を示す。この場合、コピーされたClip AV stream fileの各ソースパケットのTP_extra_headerのATS(アライ

バルタイムスタンプ)は、コピー元と同じであり、また、コピーされたClip AV stream fileに対応するClip Information fileとPlayListファイルもコピー先へコピーされる。

【0372】また、図62は、PlayList fileとClip Information fileのファイルを、コピー元(図71の再生装置2)からコピー先(図71の記録装置3)へ、IEEE1394のデジタルバス(図71のデジタルバス60)を経由して、Asynchronous(アシンクロナス)転送によって、ファイル転送し、AVストリームを、コピー元からコピー先へ、Isochronous(アイソクロナス)転送によって、リアルタイム転送(ストリーム転送)する場合を示す。

【0373】この場合、コピー元(図71の再生装置2)は、Clip Information fileに対応するClip AV stream fileをデジタルバス(図71のデジタルバス60)へ出力する時に、各ソースパケットのアライバルタイムスタンプに従って、トランスポートパケットを出力する(図7の再生モデルを参照)。各ソースパケットのTP_extra_header(ATS)は、コピー先(図71の記録装置3)で新たに付加される。また、コピーされたClip AV stream fileに対応するClip Information fileとPlayListファイルもコピー先へコピーされる。

【0374】図61または図62の場合、コピー元のPlayListとClipの内容を全て、コピー先へ転送することができるので、有効である。すなわち、コピー元のPlayListにセットされていた再生指定情報、UIAppInfoPlayList、PlayListMark、サムネール情報の内容など、また、コピー元のClipにセットされていたCPI、SequenceInfo、ProgramInfo、ClipMark、サムネール情報の内容などを、コピー先へ転送することができるので、有効である。

【0375】一方、図63、図64は、AVストリームだけをデータ転送する場合を示す。

【0376】図63は、AVストリームだけを、コピー元(図71の再生装置2)からコピー先(図71の記録装置3)へ、IEEE1394のデジタルバス(図71のデジタルバス60)を経由して、Isochronous(アイソクロナス)転送によって、リアルタイム転送(ストリーム転送)する場合を示す。この場合、コピー元は、PlayListが指定する再生区間のAVストリームをデジタルバスへ出力する時に、各ソースパケットのアライバルタイムスタンプに従って、トランスポートパケットを出力する(図7の再生モデルを参照)。コピー先の側から見ると、デジタル放送のトランスポートストリームを記録する場合と同じ状態であり、入力されたAVストリームが新たに、Clipとして記録される。すなわち、各ソースパケットのTP_extra_header(ATS)は、コピー先で新たに付加される。また、当該Clipの再生範囲をカバーするReal PlayListファイルが新たに作られる。

【0377】また、図64は、コピー先の記録装置(図71の記録装置3)がDVRフォーマットに準拠したものでない場合に、AVストリームだけをリアルタイムデータを再生する速度でデータ転送する場合を説明する図である。この場合も、図63の場合と同様にして、AVストリームを、コピー元からコピー先へ、IEEE1394のIsochronous(アイソクロナス)転送によって、リアルタイム転送(ストリーム転送)する。コピー先では、その記録装置(図71の記録装置3)のフォーマットを用いて、入力されたAVストリームを記録する。例えば、コピー先の記録装置が、D-VHS(商標)の場合は、そのフォーマットでAVストリームを記録する。このように、コピー元のDVRフォーマットに準拠した再生装置(図71の再生装置2の制御部17)が、コピー先の記録装置(図71の記録装置3の制御部17-2)と相互認証を行い、前記記録装置がDVRフォーマットに準拠していないことをわかると、前記再生装置(図71の再生装置2の制御部17)は、AVストリームをリアルタイム転送する。

【0378】図63または図64の場合、コピー元のPlayListとClipの内容を全て、コピー先へ転送することができないので、問題がある。すなわち、コピー元のPlayListにセットされていた再生指定情報、UIAppInfoPlayList、PlayListMark、サムネール情報の内容など、また、コピー元のClipにセットされていたCPI、SequenceInfo、ProgramInfo、ClipMark、サムネール情報の内容などを、コピー先へ転送することができない。一方、図61または図62の場合は、コピー元のPlayListとClipの内容を全て、コピー先へ転送することができる。

【0379】コピー先の記録装置が、本実施の形態で説明しているDVRフォーマットに準拠している場合、図61と図62のどちらでも使用することができる。

【0380】図61の方法は、AVストリームをファイル転送するので、リアルタイムにデータを再生する速度でデータ転送する場合よりも高速にデータを転送することができる。しかし、コピー先の記録装置で入力されるデータを、リアルタイムにデコードして再生することは難しい。

【0381】一方、図62の方法は、AVストリームをリアルタイムにデータを再生する速度でデータ転送するので、コピー先の記録装置で入力されるデータをリアルタイムにデコードして再生することができる。しかし、AVストリームのコピーにかかる時間は、リアルタイムにデータを再生する場合と同じ時間が必要である。図61と図62の方法は、上記のように目的別に切り替えて使用するものである。

【0382】図61と図62の例は、PlayListが一つのClipの再生範囲の全体をカバーしている場合であるが、図46で説明したように、PlayListが指定する再生範囲は、一つのClip AVストリームの全体をカバーするとは

限らない。これは、Real PlayListおよびVirtual PlayListのどちらでも、そうである。PlayListをコピー先へ転送するときは、PlayListの再生に必要なAVストリーム部分とそれが参照するClipのデータだけを、コピー先へ転送するようにした方が良い。すなわち、図65の例のように、コピー元（図72の再生装置2）から、コピー先（図71の記録装置3）へ、PlayListとそれに必要なClipの部分だけをコピーする。このようにすれば、コピー先の記録媒体（図71の記録装置3の記録媒体10-2）上に必要な空き記録容量を小さくできる。

【0383】コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、PlayListをコピーする場合に、そのPlayListの再生に必要なClip AVストリームの部分を決める方法について説明する。

【0384】図66は、あるPlayListが、オリジナルのAVストリームファイルの部分的な再生範囲を指示している時に、そのPlayListの再生に必要なストリーム部分を示す。

【0385】当該PlayListは、オリジナルAVストリーム上のIN_timeとOUT_timeを指しているとする。この場合、PlayListの再生に必要なストリーム部分は、図に示すようにソースパケット番号のX番目からY番目までである。以下の説明では、このX点とY点を決める方法の例を説明する。

【0386】図67は、AVストリームの内容を解析することをしないで、IN点の前のデータのコピー開始点（X点）を決める方法を説明する図である。PlayListはオリジナルAVストリーム上のIN点を指す。また、そのAVストリームのEP_mapを図示する。IN点が指すピクチャをデコードするためには、アドレスISA2から開始するIピクチャが必要である。また、X点の後で、PAT, PMTおよびPCRパケットが必要である。SPN_EP_start=ISA1のPTSはpts1であり、SPN_EP_start=ISA2のPTSはpts2である。pts1とpts2のシステムタイムベースの時間差が100 msec以上ならば、アドレスISA1とISA2の間にはPAT, PMTおよびPCRパケットが存在する（少なくとも、SESF, DVB, ATSC, ISDBの場合はそうである）。したがって、X点はアドレスISA1の前に決められる。そして、X点はアラインドユニットの境界でなければならない。

【0387】コピー元の再生装置は、AVストリームの内容を解析することをしないで、X点をEP_mapを使用して次のステップで決めることができる。

【0388】（S1）システムタイムベース上でIN_timeのPTSに最も近く、かつそれよりも過去の表示時刻のPTSの値を持つSPN_EP_startを見つける。

（S2）ステップS1で見つけたSPN_EP_startのPTSの値よりも少なくとも100 msec過去の表示時刻のPTSの値を持つSPN_EP_startを見つける。

（S3）X点は、ステップS2で見つけたSPN_EP_startよりも前に決められる。そして、X点はアラインドユニ

ットの境界でなければならない。

【0389】この方法は、X点を決めるためにAVストリームのデータを読み出し、その内容を解析することを必要としないので、簡単である。しかし、X点より後のAVストリームは、そのPlayListの再生には不要なデータを残してしまう場合がある。もし、X点を決めるためにAVストリームのデータを読み出し、その内容を解析するならば、そのPlayListの再生には不要なデータをより効率良く除去できる。

【0390】図68は、AVストリームの内容を解析することをしないで、OUT点の後のデータのコピー終了点（Y点）を決める方法を説明する図である。PlayListはオリジナルAVストリーム上のOUT点を指す。また、そのAVストリームのEP_mapを図示する。SPN_EP_start=ISA4から開始するビデオシーケンスは次に示すものであることを前提とする。

I2 B0 B1 P5. ...

【0391】ここで、I, P, BはそれぞれIピクチャ、PピクチャそしてBピクチャを表す。数字は表示順序を表す。この処理において、記録装置がAVストリームの内容を解析しない場合、動画像記録再生装置1（図71の記録装置3）は、OUT_timeのPTSが参照するところのピクチャの情報（ピクチャコーディングタイプ、テンポラル・レフアレンスなど）がわからない。OUT_timeのPTSはピクチャB0またはB1を参照しているかもしれない（コピー元の再生装置（図71の再生装置2）がAVストリームの内容を解析しない場合、このことはわからない）、この場合、ピクチャB0, B1をデコードするためにはI2が必要である。I2のPTSはOUT_timeのPTSよりも大きい（OUT_time < pts4, ここでpts4はI2のPTSである）。I2のPTSはOUT_timeのPTSよりも大きいが、B0, B1のためにI2が必要である。

【0392】したがって、Y点は図に示すアドレスISA5の後ろに決められる。ISA5は、EP_mapの中でISA4の直後にあるSPN_EP_startの値である。Y点はまたアラインドユニットの境界でなければならない。

【0393】コピー元の再生装置（図71の再生装置2）は、AVストリームの内容を解析することをしないで、Y点をEP_mapを使用して次のステップで決めることができる。

（S1）システムタイムベース上でOUT_timeのPTSに最も近く、かつそれよりも未来の表示時刻のPTSの値を持つSPN_EP_startを見つける。

（S2）ステップS1で見つけたSPN_EP_startの直後にあるSPN_EP_startを見つける。

（S3）Y点は、ステップS2で見つけたSPN_EP_startよりも後ろに決められる。そして、Y点はアラインドユニットの境界でなければならない。

【0394】この方法は、Y点を決めるためにAVストリームのデータを読み出し、その内容を解析することを

必要としないので、簡単である。しかし、Y点より前のAVストリームは、そのPlayListの再生には不要なデータを残してしまう場合がある。もし、Y点を決めるためにAVストリームのデータを読み出し、その内容を解析するならば、そのPlayListの再生には不要なデータをより効率良く除去できる。

【0395】次に、前記のように、コピー元から、コピー先へ、PlayListをコピーする場合に、そのPlayListの再生に必要なClip AVストリームを作成した場合の、Clip InformationファイルとPlayListの関係について説明する。

【0396】図69は、コピー元から、コピー先へ、PlayListをコピーする場合の例であり、そのPlayListの再生に必要なClip AVストリームを作成したときの、ClipとPlayListの関係を説明する図である。ここで、ClipのCPIはEP_mapであるとする。オリジナルのClipは1個のATC-sequenceと3個のSTC-sequenceを持つとする。このATC-sequenceについてのoffset_STC_id[0]はゼロである。そして、Clipの中で、stc_id=0のSTC-sequenceは、PlayItem1に使われており、stc_id=1であるSTC-sequenceは、PlayItem2とPlayItem3に使われており、stc_id=2であるSTC-sequenceは、PlayItem4に使われているとする。

【0397】今、このPlayListを別の記録媒体にコピーしたとする。すなわち、これにより、図69に示したように、このPlayListの再生に必要なClipの部分から成るClipが作られる。このように、コピー元において、1個のClip AVストリームから取り出された一つ以上のストリーム部分は、コピー先でも1個のClip AVストリームにコンパインされる。そして、新たに作られたClipは4個のATC-sequenceを持ち、それぞれのATC-sequenceは1個のSTC-sequenceを持つ。

【0398】stc_id=0であったSTC-sequenceが含まれるATC-sequenceのoffset_STC_id[0]はゼロにセットされる。また、stc_id=1であったSTC-sequenceは2個のSTC-sequenceに分かれる。1番目のATC-sequenceについてのoffset_STC_id[0]はゼロにセットされ、2番目のATC-sequenceについてのoffset_STC_id[1]は1にセットされる。すなわち、1番目のATC-sequenceの中で最後のSTC-sequenceのstc_idと2番目のATC-sequenceの中で最初のSTC-sequenceのstc_idは、同じ値で1になる。また、stc_id=2であったSTC-sequenceが含まれるATC-sequenceのoffset_STC_id[0]は2にセットされる。

【0399】これにより、コピーするPlayListのPlayItem1, PlayItem2, PlayItem3とPlayItem4のref_to_STC_idの値を変更する必要がない。あるPlayListを別の記録媒体にコピーする場合、このPlayListの再生に必要なClipの部分から成るClipを作成したときに、PlayListの内容については何も変更しなくても良い。

【0400】このように、Clip AVストリームの中にATC

の不連続点を作ることができるので、PlayListの再生に必要なClipのストリームデータを部分的に取り出す場合に、Clipファイルを分割する必要がない。さらに、ATCシーケンス毎に、その上にある最初のSTCシーケンスのSTC_idに対するoffset_STC_idを用いることにより、PlayListの再生に必要なClipの部分から成るClipを作成したときに、PlayListの内容については何も変更しなくても良い。

【0401】次に、前記のように、Clipを部分的にコピーする場合のClipの変更方法について説明する。図70は、Clipを部分的にコピーする場合の例であり、その時のSequenceInfo, ProgramInfo, CPI(EP_map)およびClipMarkの変更方法を説明する図である。

【0402】図70 (A) に示すClipが、コピー元に記録されているとする。このClipは、1個のATC-sequenceを持ち、その上にそれぞれ2個のSTC-sequenceとprogram-sequenceがある。1番目のSTC-sequenceとprogram-sequenceの開始アドレスは同じであり、それはClip AVストリーム上で0のソースパケット番号である。2番目のSTC-sequenceとprogram-sequenceの開始アドレスは同じであり、それはClip AVストリーム上でBなるソースパケット番号である。

【0403】今、図70 (A) のソースパケット番号A (A < B) から始まる付影部分のストリームデータをコピーするとする。この付影部分は、AVストリームのIN_timeからOUT_timeの再生区間に必要なストリーム部分である。コピー先に転送されるClipを図70 (B) に示す。コピー先に転送されるClipは、1個のATC-sequenceを持ち、その上にそれぞれ2個のSTC-sequenceとprogram-sequenceがある。1番目のSTC-sequenceとprogram-sequenceの開始アドレスは同じであり、それはClip AVストリーム上で0 (A - A) のソースパケット番号である。2番目のSTC-sequenceとprogram-sequenceの開始アドレスは同じであり、それはClip AVストリーム上で (B - A) なるソースパケット番号である。

【0404】図70 (C) は、図70 (A) に示すClipのClip Information fileの内容を示す。上記の説明のように、

SPN_ATC_start[0] = 0
SPN_STC_start[0] = 0, SPN_STC_start[1] = B
SPN_program_sequence_start[0] = 0, SPN_program_sequence_start[1] = Bである。また、このClip Information fileが、図示するようなEP_mapとClipMarkを持つとする。この中で、図70 (A) の付影部分のClipが使用するEP_mapとClipMarkのデータ部分は次の範囲である。EP_mapにおいては、

IN_time < pts(xa), pts(xz) < OUT_time
のPTSの値を持つエントリーポイントのデータが、図70 (A) の付影部分のClipに必要である。ClipMarkにおいては、

IN_time<pts(F), pts(Q)<OUT_time
のPTSの値を持つマークのデータが、図70 (A) の付影部分のClipに必要である。

【0405】図70 (D) は、図70 (B) に示すClipのClip Information fileの内容を示す。上記の説明のように、

```
SPN_ATC_start[0] = 0
SPN_STC_start[0] = 0, SPN_STC_start[1] = B-A
SPN_program_sequence_start[0] = 0, SPN_program_sequence_start[1] = B-Aである。また、EP_mapのデータは、図70 (C) において斜線部分のClipが使用するEP_mapのデータを元にして作成される。すなわち、エントリーポイントのPTSの値は同じであり、ソースポケット番号の値は、コピー元のClipのコピーの開始ソースポケット番号Aが差し引かれる。また、ClipMarkのデータは、図70 (C) において付影部分のClipが使用するClipMarkのデータがそのままコピーされる。
```

【0406】図71は、コピー元（出力側）の再生装置2から、コピー先（入力側）の記録装置3へ、AVストリームとそれに関するデータベースを転送する場合、および、AVストリームだけをデータ転送する場合の構成を示す。図中で、図44と同じブロック番号がついているものは同じものであることを示す。また、記録装置3の記録媒体10-2と制御部17-2は、それぞれ記録媒体10と制御部17と同じ働きをするものである。

【0407】はじめに、AVストリームとそれに関するデータベースを共に転送する場合を説明する。

【0408】まず、再生装置2の記録媒体10に記録されている所望のPlayListを記録装置3の記録媒体10-2へコピーすることを指示する情報が、図示しないユーザーインターフェースを通して、コピー制御コマンドに乗って、再生装置2へ入力される。このコマンドは、デジタルバスインターフェース50、バスコントローラ52を経由して、制御部17へ入力される。

【0409】制御部17は、上記PlayListの再生に必要なAVストリームのストリーム部分を決定し、そのAVストリームデータを記録媒体10から読み出すように、読み出し部11に指示を出す（図66、図67、図68参照）。また、制御部17は、上記PlayListに関するデータベースファイル（PlayListファイル、Clip Information fileとサムネールファイル）を記録媒体10から読み出すように、読み出し部11に指示を出す。

【0410】制御部17は、復調部12、ECC復号部13を経て読み出された前記AVストリームデータを、スイッチ61の接点A側を通して、AVストリームファイルとして、デジタルバスインターフェース50へ供給するように指示する（図61の場合）。ここで、1個のClip AVストリームファイルから取り出された一つ以上のストリーム部分は、1個のClip AVストリームファイルにコンパインされる（図69参照）。

【0411】または、制御部17は、復調部12、ECC復号部13を経て読み出された前記AVストリームデータを、スイッチ61の接点I側を通して、ソースデパケットタイザ14へ入力しても良い（図62の場合）。この場合、ソースデパケットタイザ14は、アライバルタイムスタンプに従って、トランスポートストリームをデジタルバスインターフェース50へ供給する。

【0412】一方、記録媒体10から読み出されたAVストリームデータに対応するデータベースファイルが、復調部12、ECC復号部13を経て、メモリ51へ入力される。制御部17は、メモリ51にあるデータを元にして、デジタルバスインターフェース50から出力される上記のAVストリームファイルの再生に必要なデータベース（Clip Informationファイル、PlayListファイル）を作成する（図69、図70参照）。また、制御部17は、上記のAVストリームファイルに対応するClipとコピーするPlayListファイルが使用するサムネールファイルを作成する（後述の図72のステップS172参照）。

【0413】そして、制御部17は、上記新たに作成されたデータベースファイル（Clip Informationファイル、PlayListファイルとサムネールファイル）をメモリ51からデジタルバスインターフェース50へ供給するように指示する。

【0414】バスコントローラ52は、デジタルインターフェース50からのファイル出力を制御する。制御部17は、バスコントローラ52に対して、デジタルバスインターフェース50から、AVストリームとそれに関するデータベースを出力するように指示する。

【0415】前記AVストリームとそれに関するデータベースは、デジタルバス60を経由して、コピー先の記録装置3へ入力される。

【0416】コピー先の記録装置3のバスコントローラ57は、デジタルバスインターフェース55からのファイル入力を制御する。また、バスコントローラ52とバスコントローラ57は、ファイルのコピー制御コマンドを交換して、データ伝送／受信のタイミングを制御する。

【0417】記録装置3の制御部17-2は、コピー元からデジタルバスインターフェース55へ、AVストリームファイルとして入力される場合、スイッチ62の接点A側を通して、ECC符号化部30、変調部31、書き込み部32の処理を経て、記録媒体10-2に記録するように指示する。

【0418】または、制御部17-2は、コピー元からデジタルバスインターフェース55へ、トランスポートストリームとして入力される場合、スイッチ62の接点I側を通して、ソースパケットタイザ29へ入力する。ソースパケットタイザ29は、トランスポートパケットをソースパケット化して出力する。制御部17-2は、ソースパ

ケットから成るAVストリームを、ECC符号化部30、変調部31、書き込み部32の処理を経て、記録媒体10-2に記録するように指示する。

【0419】また、制御部17-2は、ディジタルバスインターフェース55へ入力されるデータベースファイルを、メモリ56へ書きこむように指示する。

【0420】また、制御部17-2は、記録媒体10-2に記録されているデータベースファイル(Info.dvrファイルとサムネールファイル)を、読み出し部11、復調部12、ECC復号部13の処理を経て、メモリ56へ読み出すように指示する。 10

【0421】そして、制御部17-2は、メモリ56にあるInfo.dvrファイルとサムネールファイルを更新する。具体的には、コピー先にあるInfo.dvrファイルのTableOfPlayListに、新たに記録するPlayListファイル名を追加し、また、コピー先にあるサムネールファイルに新たに記録するサムネールを追加する(後述の図74のステップ224、S225参照)。

【0422】制御部17-2は、メモリ56にあるデータベースファイルを読み出して、ECC符号化部30、変調部31、書き込み部32の処理を経て、記録媒体10-2に記録するように指示する。 20

【0423】このようにして、再生装置2から、記録装置3へ、AVストリームとそれに関するデータベースをデータ転送してコピーする場合の処理を行う。

【0424】次に、再生装置2がコピー先へAVストリームだけを転送する場合を説明する。これは、コピー先の記録装置3が、DVRフォーマットに準拠していない場合における、再生装置2のトランスポートストリームの再生動作と考えられる。

【0425】制御部17は、上記PlayListの再生に必要なAVストリームのストリーム部分を決定し、そのAVストリームデータを記録媒体10から読み出すように、読み出し部11に指示を出す。制御部17は、復調部12、ECC復号部13を経て読み出された前記AVストリームデータを、スイッチ61の接点I側を通して、ソースデパケットタイザ14へ入力する。ソースデパケットタイザ14は、アライバルタイムスタンプに従って、トランスポートパケットをディジタルバスインターフェース50へ供給する。ディジタルバスインターフェース50は、トランスポートパケットをアイソクロナス転送する。 30

【0426】また、記録装置3へAVストリームだけが転送される場合を説明する。これは、コピー元の再生装置2が、DVRフォーマットに準拠していない場合における、記録装置3のトランスポートストリームの記録動作と考えられる。

【0427】記録装置3の制御部17-2は、ディジタルバスインターフェース55へ入力されるトランスポートストリームを、スイッチ62の接点I側を通して、ソースデパケットタイザ29へ入力する。ソースデパケットタイザ29 50

は、トランスポートパケットをソースパケット化して出力する。制御部17-2は、ソースパケットから成るAVストリームを、ECC符号化部30、変調部31、書き込み部32の処理を経て、記録媒体10-2に記録するように指示する。

【0428】また、スイッチ62の接点I側を通して、トランスポートストリームは多重化ストリーム解析部26へ入力される。ここでの処理の内容は、前述の図44で説明したとおりである。制御部17-2は、解析部26での解析結果に基づいて、データベースファイルを作成する。制御部17-2は、データベースファイルをECC符号化部30、変調部31、書き込み部32の処理を経て、記録媒体10-2に記録するように指示する。 10

【0429】このようにして、再生装置2からAVストリームだけをデータ転送する場合の処理、また、再生装置3へAVストリームだけがデータ転送される場合の処理を行う。

【0430】図72は、コピー元(出力側の再生装置2)から、コピー先(入力側の記録装置3)へ、PlayListをコピーする場合の、コピー元の制御部17が行う処理を説明するフローチャートである。

【0431】ステップS170で、AVストリームとそれに関するデータベースをデータ転送してコピーする場合と、AVストリームだけをデータ転送してコピーする場合かを決める。前者の場合は、ステップS171へ進む。後者の場合は、ステップS175へ進む。

【0432】ステップS171で、Clipについての次の処理を行う。

- 30 · 当該PlayListの再生に必要なClip AVストリームの部分を決定する(図66、図67、図68参照)。
- 上記決定したAVストリーム部分が使用するClip Informationファイルを作成する(図70参照)。

【0433】ステップS172で、サムネールファイルについての次の処理を行う。

- 30 · 当該PlayListが使用するメニュー・サムネールを含むファイルを作成する。
- 当該PlayListおよび上記決定したAVストリーム部分が、使用するマーク・サムネールを含むファイルを作成する。

【0434】ステップS173で、AVストリームファイルの転送の処理を行う。

- 上記Clipの処理で決定したストリーム部分のAVストリームデータをコピー先へ転送する(図69参照)。

【0435】ステップS174で、データベースファイルの転送の処理を行う。

- 上記PlayListファイルをコピー先へ転送する。
- 上記Clipの処理で作成したClip Informationファイルをコピー先へ転送する。
- 上記ステップS172で作成したサムネールファイルをコピー先へ転送する。

【0436】ステップS170で、AVストリームだけをデータ転送してコピーすると判定され、ステップS175へ進む場合、図60に示すPlayListの再生を説明するフローチャートで読み出されるAVストリームデータ(S135で読み出すAVストリームデータ)を、トランスポートストリーム化して、コピー先へ転送する。

【0437】図73は、上記ステップS171のClipについての処理の詳細を説明するフローチャートである(図70参照)。

【0438】ステップS201で、PlayListが、当該Clipの中で使用する再生区間の表示開始時刻および表示終了時刻を取得する。

【0439】ステップS202で、上記時間区間に応するClip AVストリーム上のコピー開始パケット(アドレス)とコピー終了パケット(アドレス)をCPIに基づいて決定する。

【0440】ステップS203で、CPIについての次の処理を行う。

- ・コピーする区間のAVストリーム部分が使用するCPIエントリーを取得する。
- ・CPIのエントリーポイントのソースパケット番号の値を、コピーするAVストリーム部分の中でのソースパケット番号の値へ変更する。

【0441】ステップS204で、SequenceInfoについての次の処理を行う。

- ・コピーする区間のAVストリーム部分におけるATC-sequenceの開始パケット番号を更新する。
- ・コピーする区間のAVストリーム部分におけるSTC-sequenceの開始パケット番号を更新する。
- ・コピーする区間のAVストリーム部分におけるATCシーケンス上にあるSTC-sequenceに対するSTC-idの値が変わらないようにoffset_STC_idを更新する。

【0442】ステップS205で、ProgramInfoについての次の処理を行う。コピーする区間のAVストリーム部分におけるprogram-sequenceの開始パケット番号を更新する。

【0443】ステップS206で、ClipMarkについての次の処理を行う。コピーする区間のAVストリーム部分が使用するMarkエントリーを取得する。

【0444】ステップS207で、ファイル作成についての次の処理を行う。

- ・コピーする区間のAVストリーム部分をコンパインしてClip AV stream fileにする。
- ・Clip Information fileを作成する。

【0445】図74は、コピー元(出力側の再生装置2)から、コピー先(入力側の記録装置3)へ、PlayListをコピーする場合の、コピー先が行う処理を説明するフローチャートである。

【0446】ステップS221で、AVストリームとそれに関係するデータベースをデータ転送してコピーする

場合と、AVストリームだけをデータ転送してコピーする場合を決める。前者の場合は、ステップS222へ進む。後者の場合は、ステップS226へ進む。

【0447】ステップS222で、AVストリームファイルの管理の処理を行う。入力されたAVストリームデータをClip AVストリームファイルとして、STREAMディレクトリへ記録する。

【0448】ステップS223で、データベースファイルの管理についての処理を行う。

- ・入力されたPlayListファイルをPLAYLISTディレクトリへ記録する。
- ・入力されたClip InformationファイルをCLIPINFディレクトリへ記録する。

【0449】ステップS224で、Info.dvrについての処理を行う。入力されたPlayListファイルをコピー先にあるInfo.dvrのTableOfPlayListに追加する。

【0450】ステップS225で、サムネールファイルについての処理を行う。入力されたサムネールファイルにエントリーされているサムネールデータを、コピー先にあるサムネールファイルへ追加する。

【0451】ステップS221で、AVストリームだけをデータ転送してコピーすると判定され、ステップS226へ進む場合、図58に示すReal PlayListの作成を説明するフローチャートによって、コピー先に入力されるトランSPORTストリームを記録して、Real PlayListを作成する。

【0452】このようなシンタクス、データ構造、規則に基づく事により、記録媒体10に記録されているAVストリームファイルとそのデータベースファイルの内容を適切に管理することができ、もって、ユーザが所望のAVストリームファイルとそのデータベースファイルを別の記録媒体に簡単にコピーできるようにすることができます。

【0453】また、記録媒体10に記録されているAVストリームファイルの部分的な再生区間を別の記録媒体にコピーする場合に、当該再生区間の再生に必要なAVストリームファイルを簡単に作成することができ、当該再生区間の再生に必要なデータベースファイルを簡単に作成することができるので、もって、ユーザが、所望のAVストリームファイルの部分的な再生区間の再生に必要なAVストリームファイルとデータベースファイルを作成することができ、それらファイルを別の記録媒体に簡単にコピーできるようにすることができます。

【0454】なお、図71の説明では、コピー先からコピー元へのファイル転送に用いる伝送路が、IEEE1394等のディジタルバスの場合を説明したが、これに限らず、図75に示すように、放送波等の無線波を伝送路としてもよい。また、コピー制御のコマンドは、ファイル転送用の伝送路と同じである必要はない。(図75参照)

【0455】また、AVストリームファイルとそのデータ

タベースファイルが別々の記録媒体から、コピー先に転送されてもよい。図76に示すように、AVストリームファイルが記録されているサーバーとデータベースファイルが記録されているサーバーが別々に用意されており、記録装置3からのコピー制御コマンドに対して、AVストリームファイルサーバーとデータベースファイルサーバーから、ファイルが記録装置3へ転送されるようにもても良い。

【0456】なお、本実施の形態は、多重化ストリームとしてMPEG2トランSPORTストリームを例にして説明しているが、これに限らず、DSSトランSPORTストリームやMPEG2プログラムストリームについても適用することが可能である。

【0457】上述した一連の処理は、ハードウェアにより実行させることもできるが、ソフトウェアにより実行させることもできる。この場合、そのソフトウェアを構成するプログラムが、専用のハードウェアに組み込まれているコンピュータ、または、各種のプログラムをインストールすることで、各種の機能を実行することが可能な、例えば汎用のパーソナルコンピュータなどに、ネットワークや記録媒体からインストールされる。

【0458】この記録媒体は、図44に示すように、装置本体とは別に、ユーザにプログラムを提供するために配布される、プログラムが記録されている磁気ディスク51(フロッピディスクを含む)、光ディスク52(CD-ROM(Compact Disk-Read Only Memory), DVD(Digital Versatile Disk)を含む)、光磁気ディスク53(Mini-Disk)を含む)、もしくは半導体メモリ54などよりなるパッケージメディアにより構成されるだけでなく、装置本体に予め組み込まれた状態でユーザに提供される、プログラムが記録されているROMやハードディスクなどで構成される。

【0459】なお、本明細書において、記録媒体に記録されるプログラムを記述するステップは、記載された順序に沿って時系列的に行われる処理はもちろん、必ずしも時系列的に処理されなくとも、並列的あるいは個別に実行される処理をも含むものである。

【0460】また、本明細書において、システムとは、複数の装置により構成される装置全体を表すものである。

【0461】

【発明の効果】本発明によれば、シンタクス、データ構造、規則に基づくことにより、記録媒体に記録されているAVストリームファイルとそのデータベースファイルの内容を適切に管理することができ、もって、ユーザが所望のAVストリームファイルとそのデータベースファイルを別の記録媒体に簡便にコピーできるようにすることができる。

【0462】特に、請求項1のデータ伝送装置、請求項13のデータ伝送方法、請求項14の記録媒体のプログ

ラム、および請求項15のプログラムによれば、記録媒体に記録されているAVストリームファイルの部分的な再生区間を別の記録媒体にコピーする場合に、当該再生区間の再生に必要なAVストリームファイルを簡便に作成することができ、当該再生区間の再生に必要なデータベースファイルを簡便に作成ことができるので、もって、ユーザが、所望のAVストリームファイルの部分的な再生区間の再生に必要なAVストリームファイルとデータベースファイルを作成することができ、それらファイルを別の記録媒体に簡便にコピーできるようにすることができます。また、コピー元から、コピー先へ、PlayListとそれに必要なClipの部分だけをコピーするので、コピー先の記録媒体上に必要な空き記録容量を小さくできる。

【0463】請求項4乃至7、および9のデータ伝送装置によれば、コピー元のPlayListとClipの内容を全て、コピー先へ転送することができるので、有効である。すなわち、コピー元のPlayListにセットされていた再生指定情報、UIAppInfoPlayList, PlayListMark, サムネール情報の内容など、また、コピー元のClipにセットされていたCPI, SequenceInfo, ProgramInfo, ClipMark, サムネール情報の内容などを、コピー先へ転送することができるので、有効である。

【0464】請求項10のデータ伝送装置によれば、AVストリームをファイル転送するので、リアルタイムにデータを再生する速度でデータ転送する場合よりも高速にデータを転送することができる。

【0465】請求項11のデータ伝送装置によれば、AVストリームをリアルタイムにデータを再生する速度でデータ転送するので、コピー先の記録装置で入力されるデータをリアルタイムにデコードして再生することができる。

【0466】請求項12と16のデータ伝送装置、請求項17のデータ伝送方法、請求項18の記録媒体のプログラム、および請求項19のプログラムによれば、伝送先の装置が、例えば、DVRフォーマットに準拠したものである場合は、上記請求項1の伝送装置の方法でデータ伝送することができ、コピー先の記録装置がDVRフォーマットに準拠したものでない場合は、AVストリームだけをリアルタイムにデータを再生する速度でデータ転送するように、処理を切り替えることができるので、ユーザーのAVストリームのコピー操作の利便性を向上できる。

【0467】請求項20のデータ処理装置、請求項21のデータ伝送方法、請求項22の記録媒体のプログラム、および請求項23のプログラムによれば、入力されたPlayListを適切に記録媒体上で管理することができ、もって、ユーザが記録媒体にコピーされたPlayListの内容をわかりやすく確認できる。

【0468】請求項24のデータ処理装置、請求項25

のデータ伝送方法、請求項26の記録媒体のプログラム、および請求項27のプログラムによれば入力されたサムネールを適切に記録媒体上で管理することができ、もって、ユーザが記録媒体にコピーされたサムネールの内容をわかりやすく確認できる。

【画面の簡単な説明】

【図1】volume Informationを説明する図である。

【図2】ディスク上につくられるディレクトリ構造を説明する図である。

【図3】DVR MPEG-2 transport streamの構造を示す図である。

【図4】source_packetのシンタクスを示す図である。

【図5】TP_extra_header()のシンタクスを示す図である。

【図6】DVR MPEG-2 transport stream recorder modelを示す図である。

【図7】DVR MPEG-2 transport stream player modelを示す図である。

【図8】Clip Information fileのシンタクスを示す図である。

【図9】ATC-sequenceについて説明する図である。

【図10】ATCの不連続点とATC-sequenceの関係を説明する図である。

【図11】連続なSTC区間について説明する図である。

【図12】STCの不連続点とSTC-sequenceの関係、およびSTC-sequenceとATC-sequenceの関係を説明する図である。

【図13】SequenceInfo()のシンタクスを示す図である。

【図14】program_sequenceを説明する図である。

【図15】ProgramInfo()のシンタクスを示す図である。

【図16】StreamCodingInfo()のシンタクスを示す図である。

【図17】stream_coding_typeを示す図である。

【図18】video_formatを示す図である。

【図19】frame_rateを示す図である。

【図20】display_aspect_ratioを示す図である。

【図21】audio_presentation_typeを示す図である。

【図22】sampling_frequencyを示す図である。

【図23】CPI()のシンタクスを示す図である。

【図24】EP_mapを説明する図である。

【図25】AVストリームをはじめてClipとして記録するときにできるTU_mapについて説明する図である。

【図26】TU_mapのシンタクスを示す図である。

【図27】PlayListMarkとClipMarkの関係について説明する図である。

【図28】ClipMarkのシンタクスを示す図である。

【図29】PlayList fileのシンタクスを示す図である。

【図30】UIAppInfoPlayListのシンタクスを示す図である。

【図31】PlayList()のシンタクスを示す図である。

【図32】EP_map typeのPlayListについて説明する図である。

【図33】TU_map typeのPlayListについて説明する図である。

【図34】EP_map typeのPlayListの時間情報とAVストリームファイルの中のアドレス情報の関係を説明する図である。

【図35】TU_map typeのPlayListの時間情報とAVストリームファイルの中のアドレス情報の関係を説明する図である。

【図36】PlayItem()のシンタクスを示す図である。

【図37】PlayListMarkのシンタクスを示す図である。

【図38】Info.dvrのシンタクスを示す図である。

【図39】UIAppInfoVolumeのシンタクスを示す図である。

【図40】TableOfPlayListsのシンタクスを示す図である。

【図41】サムネールのヘッダ情報ファイルのシンタクスを示す図である。

【図42】サムネールのピクチャデータファイルのシンタクスを示す図である。

【図43】tn_blockへのデータの格納方法を説明する図である。

【図44】動画像記録再生装置の構成を示すブロック図である。

【図45】AVストリームが新しいClipとして記録される時のClipとPlayListの関係のコンセプトを説明する図である。

【図46】Virtual PlayListの作成のコンセプトについて説明する図である。

【図47】Real PlayListの再生区間の一部分を消去したときのClipとPlayListの関係のコンセプトを説明する図である。

【図48】Minimizeの編集をしたときのClipとPlayList, Virtual PlayListの関係のコンセプトを説明する図である。

【図49】1つのClip AVストリームのデータを部分的に消去した時に、Clipの中に2つのATC_sequencesができる場合を説明する図である。

【図50】1つのClip AVストリームのデータを部分的に消去した時のATC_sequences, STC_sequencesおよびprogram_sequenceの関係を説明する図である。

【図51】CPIがEP_mapであるClip AVストリームの一部分を消去した時のClipとPlayListの関係を説明する図である。

【図52】Clipの中にATCの不連続を許さない場合において、Clip AVストリームの一部分を消去した時に、Cli

pが2つに分かれる場合を説明する図である。

【図53】CPIがTU_mapであるClip AVストリームの一部分を消去した時のClipとPlayListの関係を説明する図である。

【図54】AVストリームをClipして新しく記録するときの、Clip AVストリームファイルおよびClip Informationファイルの作成を説明するフローチャートである。

【図55】はじめにAVストリームをClipとして記録するときのSequenceInfoの作成の動作例を説明するフローチャートである。

【図56】ProgramInfoの作成の動作例を説明するフローチャートである。

【図57】EP_mapの作成の動作例を説明するフローチャートである。

【図58】Real PlayListの作成方法を説明するフローチャートである。

【図59】Virtual PlayListの作成方法を説明するフローチャートである。

【図60】PlayListの再生方法を説明するフローチャートである。

【図61】AVストリームファイルとデータベースファイルと共にファイル転送する場合を説明する図である。

【図62】AVストリームをリアルタイム転送（ストリーム転送）し、データベースをファイル転送する場合を説明する図である。

【図63】AVストリームだけをリアルタイム転送し、コピー先でデータベースファイルを新たに作成する場合を説明する図である。

【図64】コピー先の記録装置がDVRフォーマットに準拠したものではない場合に、AVストリームをリアルタイムにデータを再生する速度で転送する場合を説明する図である。

【図65】コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、PlayListとそれに必要なClipの部分だけをコピーする場合のClipとPlayListの関係を説明する図である。

【図66】コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、PlayListをコピーする場合に、そのPlayListの再生に必要なClipの部分について説明する図である。

【図67】コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、PlayListをコピーする場合に、IN_timeの前のデータのコピー開始点を決める方法 *

*を説明する図である。

【図68】コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、PlayListをコピーする場合に、OUT_timeの後のデータのコピー終了点を決める方法を説明する図である。

【図69】コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、PlayListとそれに必要なClipの部分だけをコピーする場合のClipとPlayListの関係を説明する図である。

10 【図70】コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、Clipの部分的にコピーする場合のClipの変更方法について説明する図である。

【図71】コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、DVRのAVストリームファイルとそれに関するデータベースファイルを、デジタルバス経由でコピーする場合の構成を示すブロック図である。

20 【図72】コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、PlayListをコピーする場合の、コピー元の処理を説明するフローチャートである。

【図73】コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、PlayListをコピーする場合において、コピー元の側のClipについての処理を説明するフローチャートである。

【図74】コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、PlayListをコピーする場合の、コピー先の処理を説明するフローチャートである。

30 【図75】AVストリームファイルとデータベースファイルが、無線波を経由して、記録装置へファイルが転送される場合を説明する図である。

【図76】AVストリームファイルとデータベースファイルが、別々のサーバに記録されており、それぞれのサーバから記録媒体へファイルが転送される場合を説明する図である。

【符号の説明】

10 記録媒体, 11 読み出し部, 12 復調部, 13 ECC復号部, 14 ソースデパケットизатор, 15 デマルチプレクサ, 16 AVデコーダ, 17 制御部, 23 AVエンコーダ, 24 ビデオ解析部, 25 マルチプレクサ, 26 多重化ストリーム解析部, 29 ソースパケットизатор, 30 ECC符号化部, 31 変調部, 32 書き込み部, 61 再生部, 62 記録部

【図1】

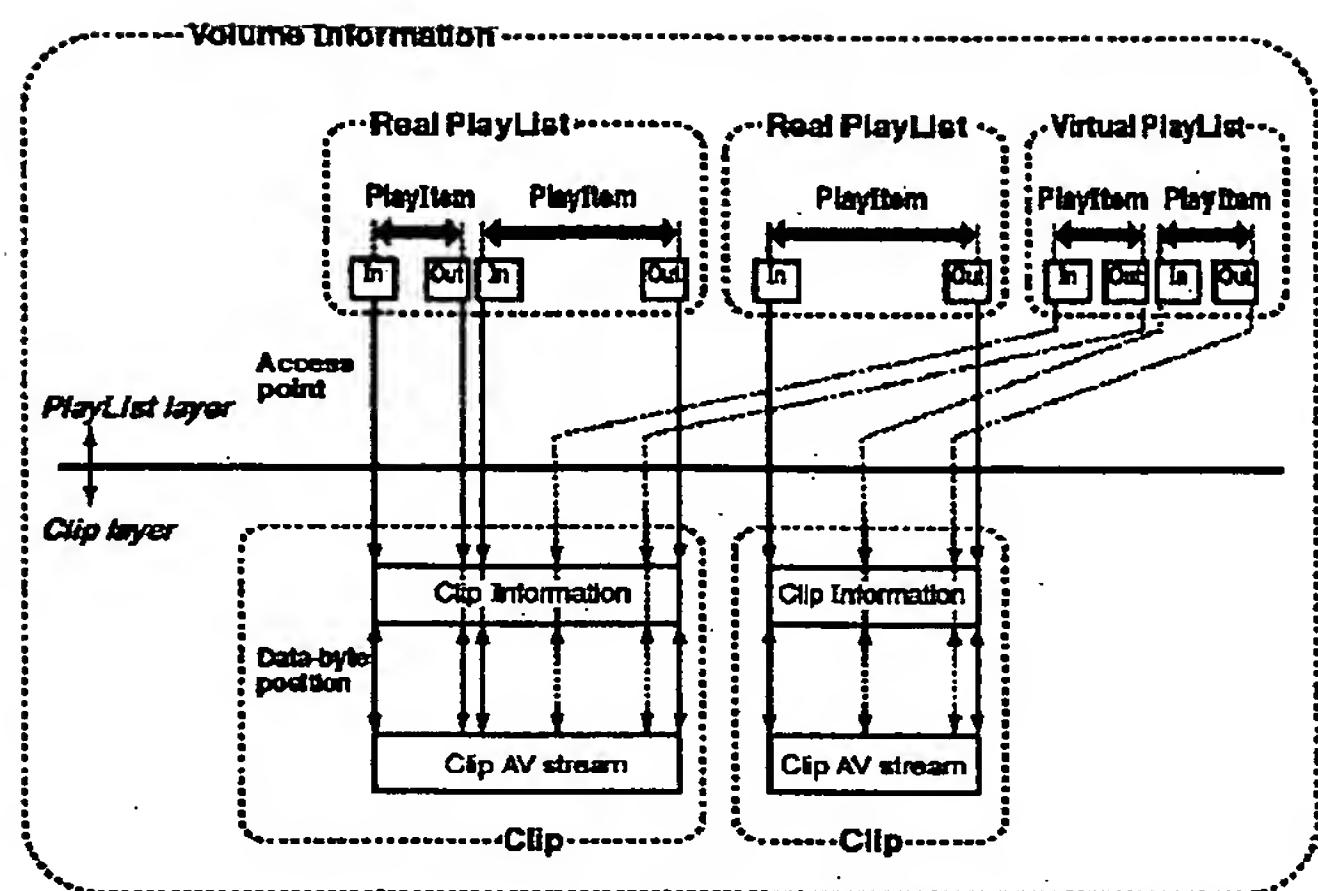
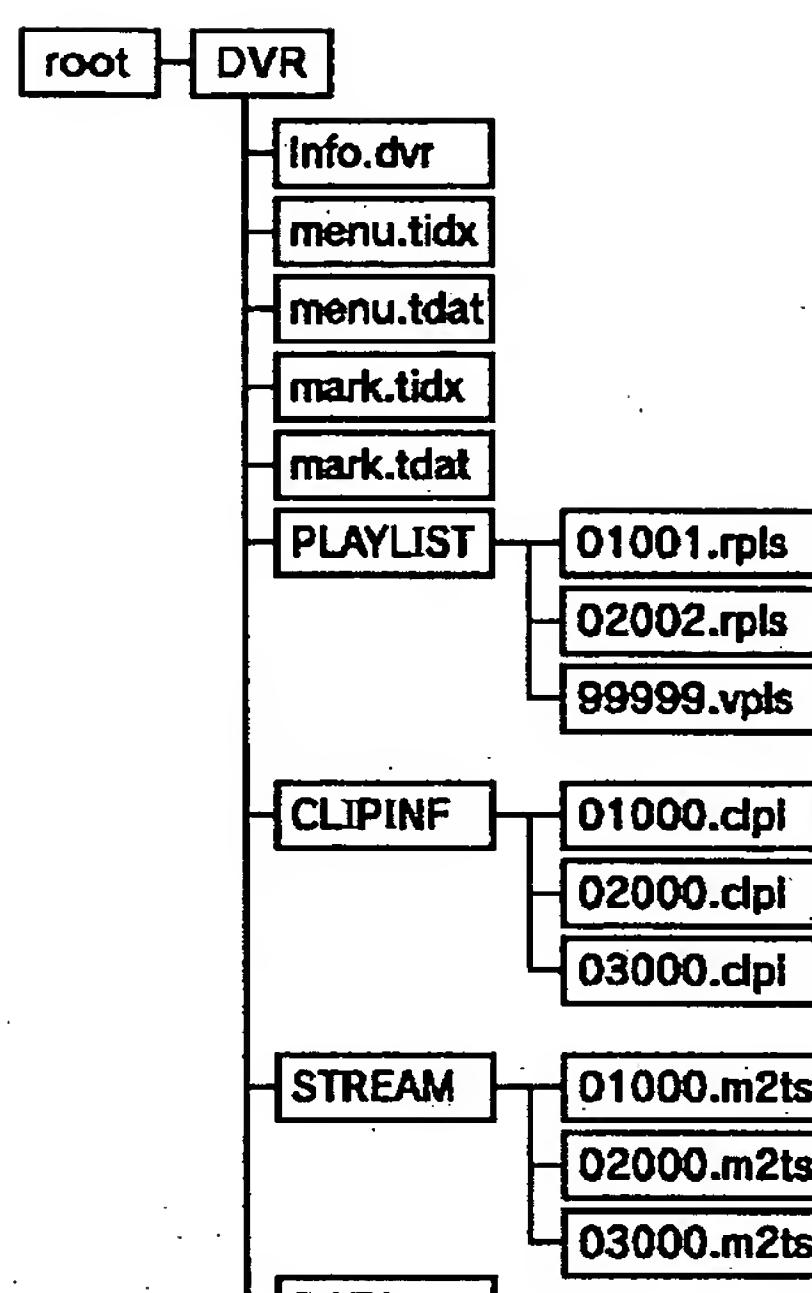


図1

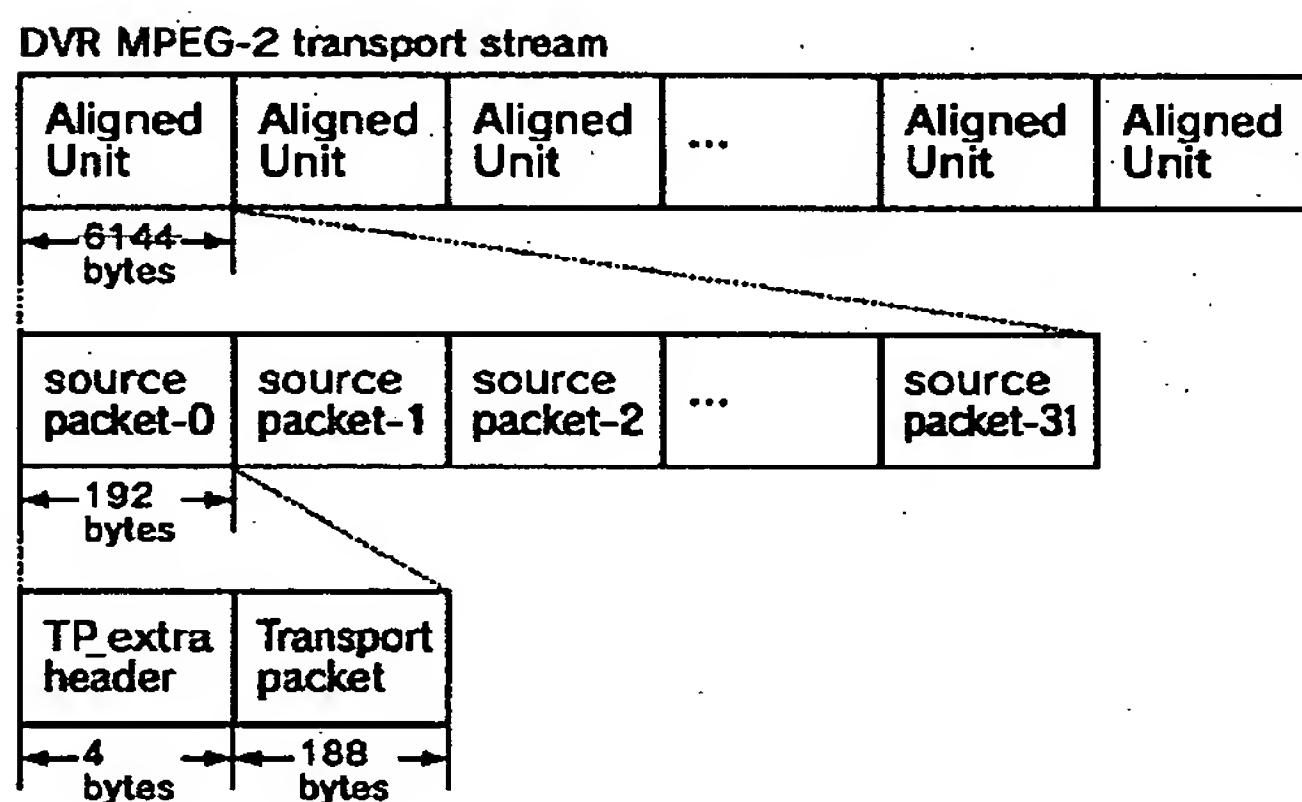
【図2】



ディスク上につくられるディレクトリ構造の例

図2

【図3】



DVR MPEG-2 transport stream の構造

図3

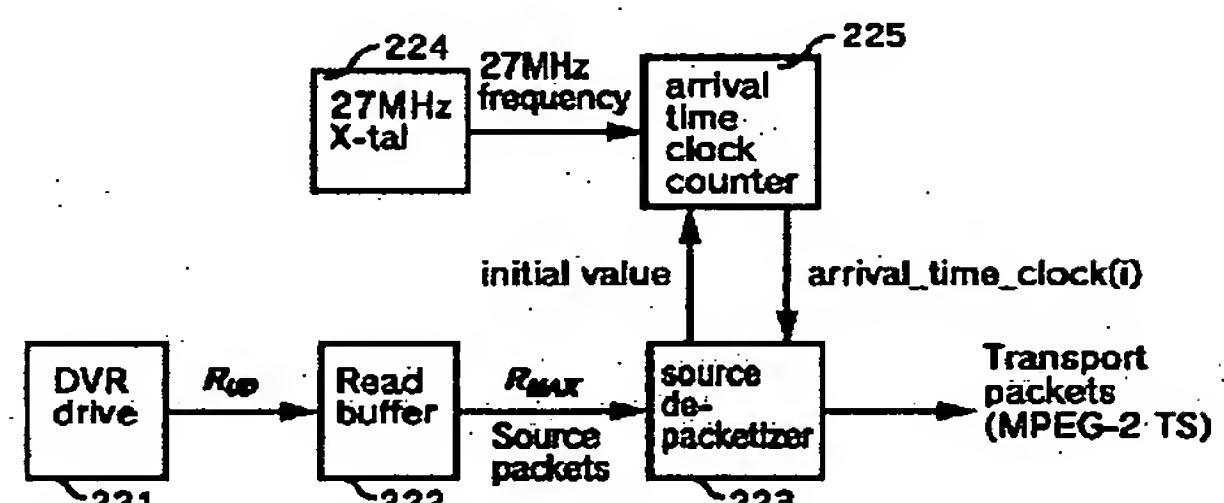
【図4】

| Syntax | No. of bits | Mnemonic |
|--------------------|-------------|----------|
| source_packet(){ | | |
| TP_extra_header() | | |
| transport_packet() | | |
| } | | |

source_packet のシンタクス

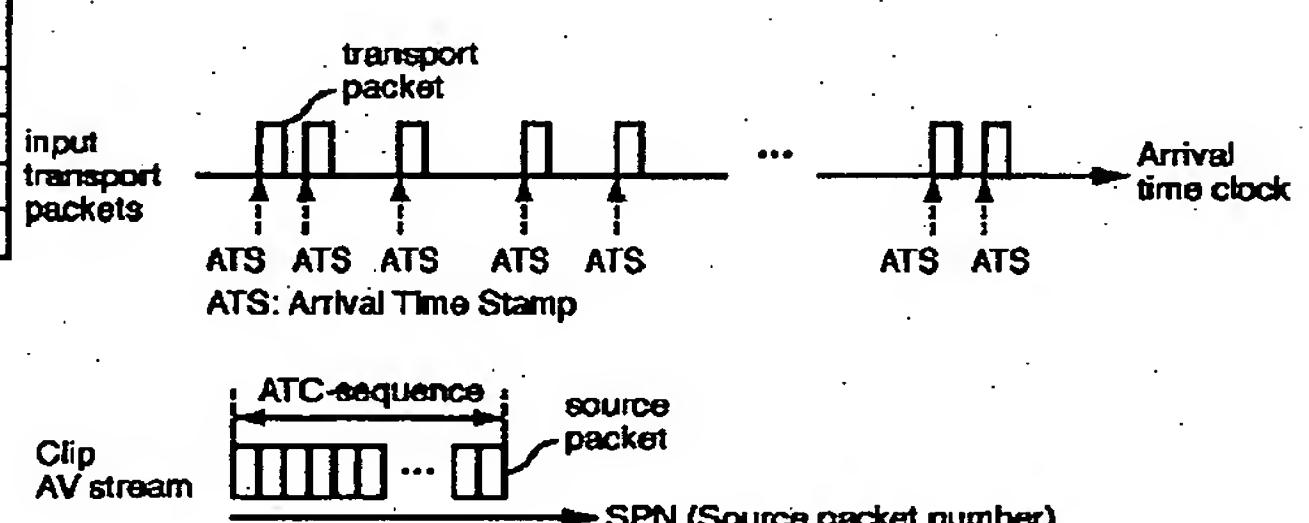
図4

【図7】



DVR MPEG-2 transport stream player model

【図9】



ATC-sequence について説明する図

【図5】

| Syntax | No. of bits | Mnemonic |
|---------------------------|-------------|----------|
| TP_extra_header() | | |
| copy_permission_indicator | 2 | uimsbf |
| arrival_time_stamp | 30 | uimsbf |
| } | | |

TP_extra_header() のシンタクス

図5

【図6】

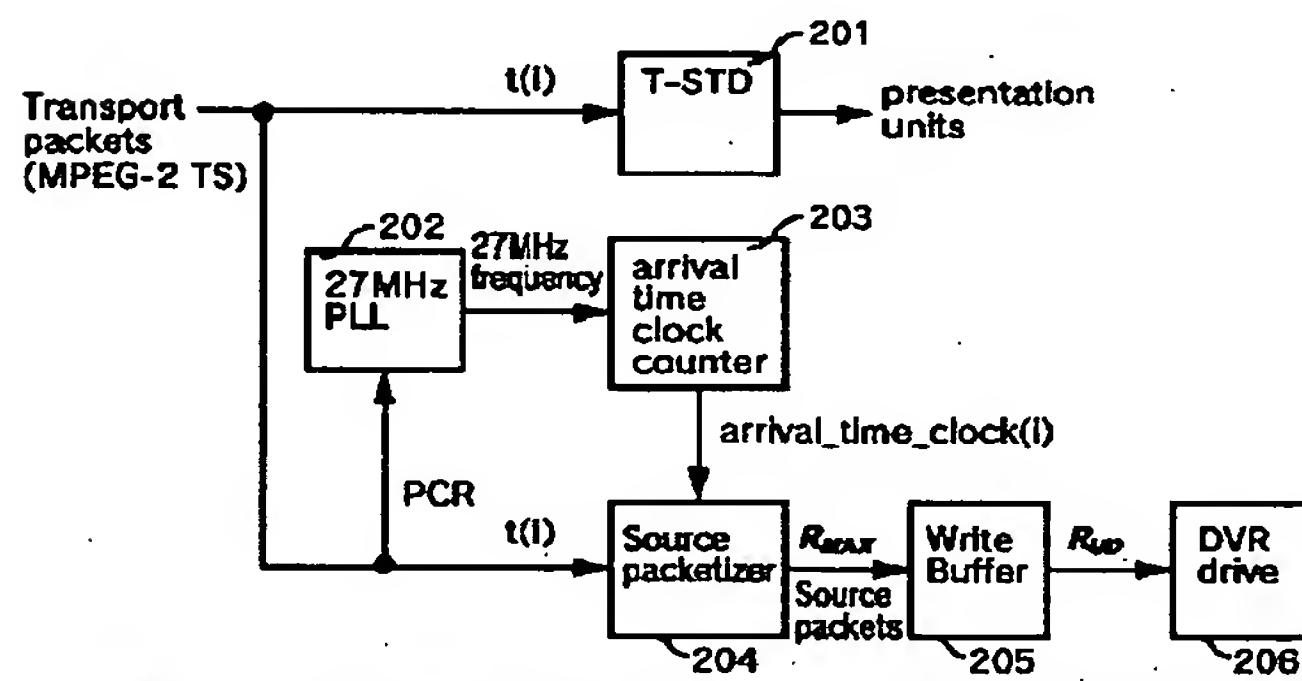


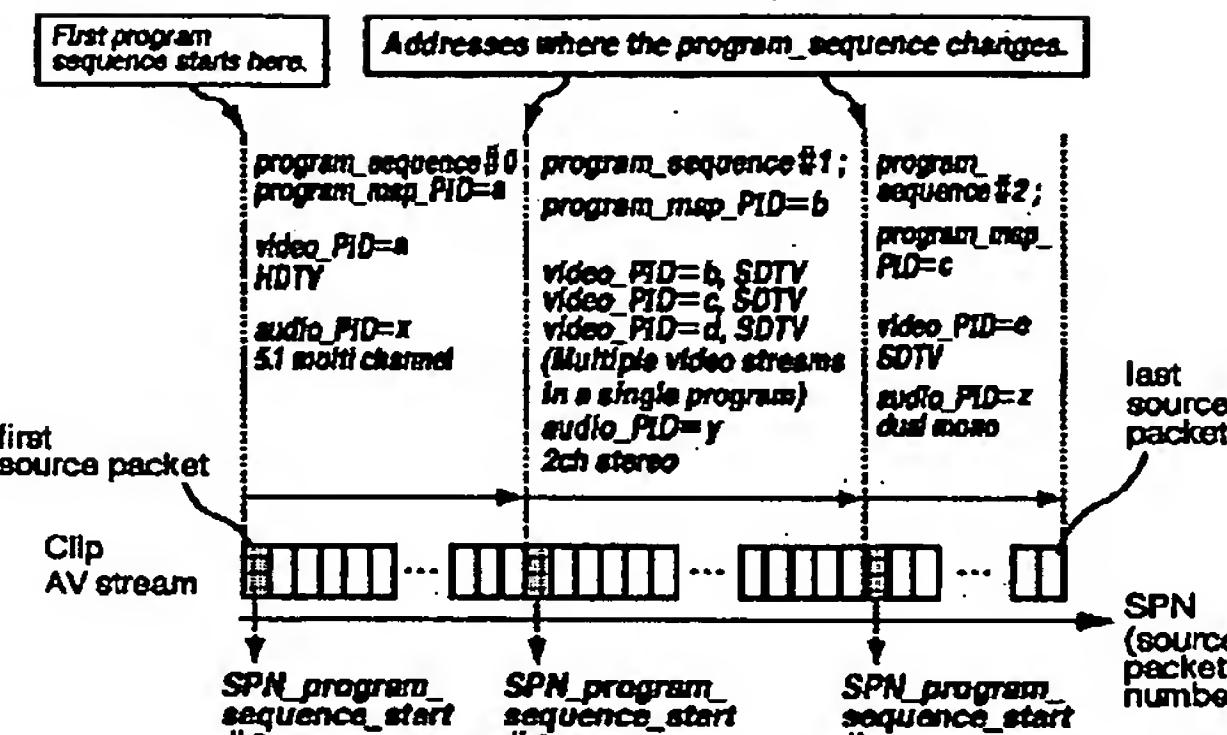
図6 DVR MPEG-2 transport stream recorder model

【図8】

| Syntax | No. of bits | Mnemonic |
|----------------------------------|-------------|----------|
| zzzzz.cpl { | | |
| version_number | 8*4 | bslbf |
| SequenceInfo_start_address | 32 | uimsbf |
| ProgramInfo_start_address | 32 | uimsbf |
| CPI_start_address | 32 | uimsbf |
| ClipMark_start_address | 32 | uimsbf |
| MarkersPrivateData_start_address | 32 | uimsbf |
| reserved_for_future_use | 96 | bslbf |
| ClipInfo() | | |
| for(i=0; i<N1; i++) | | |
| Padding_word | 16 | bslbf |
| } | | |
| SequenceInfo() | | |
| for(i=0; i<N2; i++) | | |
| Padding_word | 16 | bslbf |
| } | | |
| ProgramInfo() | | |
| for(i=0; i<N3; i++) | | |
| Padding_word | 16 | bslbf |
| } | | |
| CPI() | | |
| for(i=0; i<N4; i++) | | |
| Padding_word | 16 | bslbf |
| } | | |
| ClipMark() | | |
| for(i=0; i<N5; i++) | | |
| Padding_word | 16 | bslbf |
| } | | |
| MarkersPrivateData() | | |
| for(i=0; i<N6; i++) | | |
| Padding_word | 16 | bslbf |
| } | | |
| } | | |

図8 Clip Information file のシンタクス

【図14】



program-sequence を説明する図

図14

【図24】

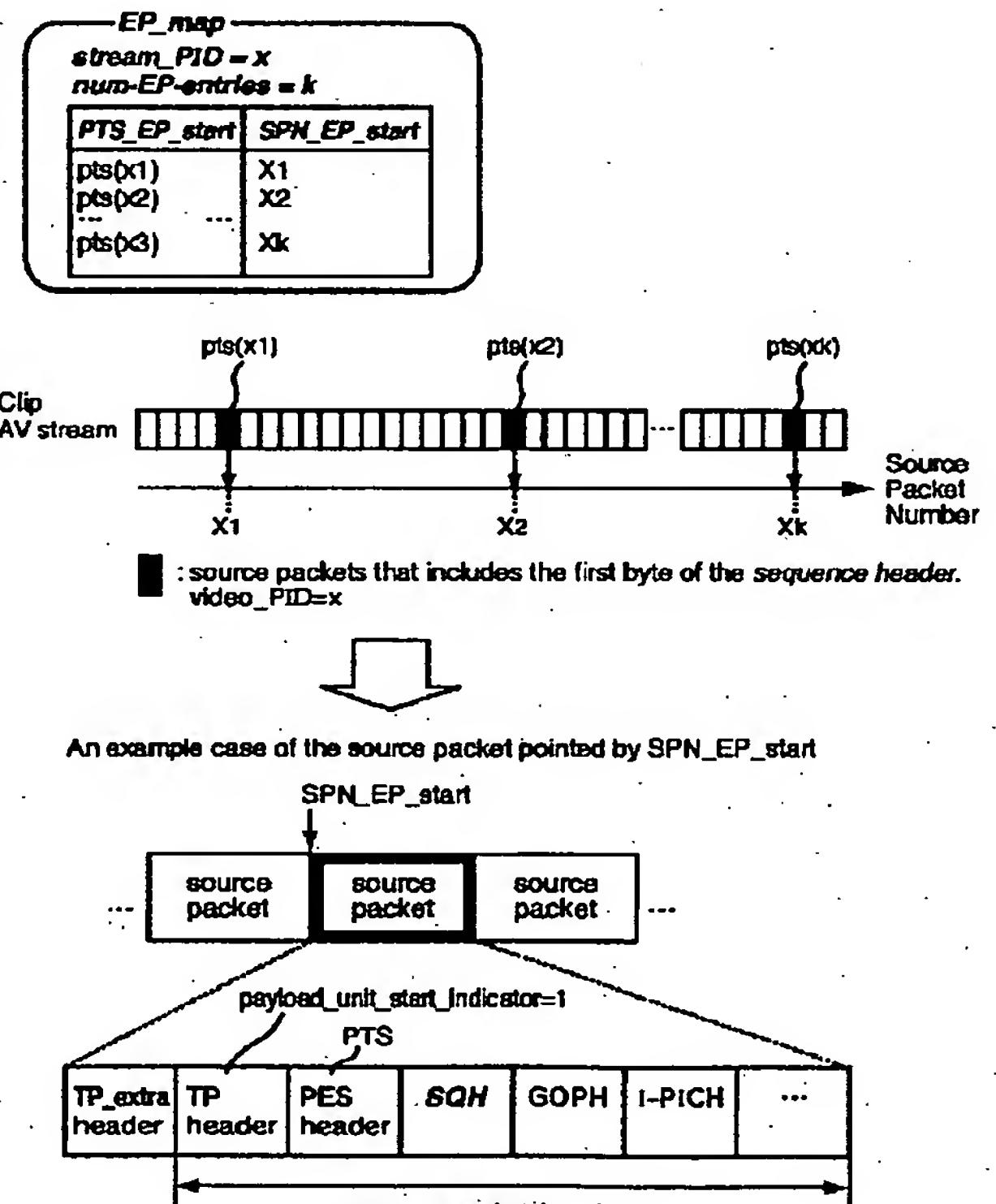


図24 EP-map を説明する図

【図10】

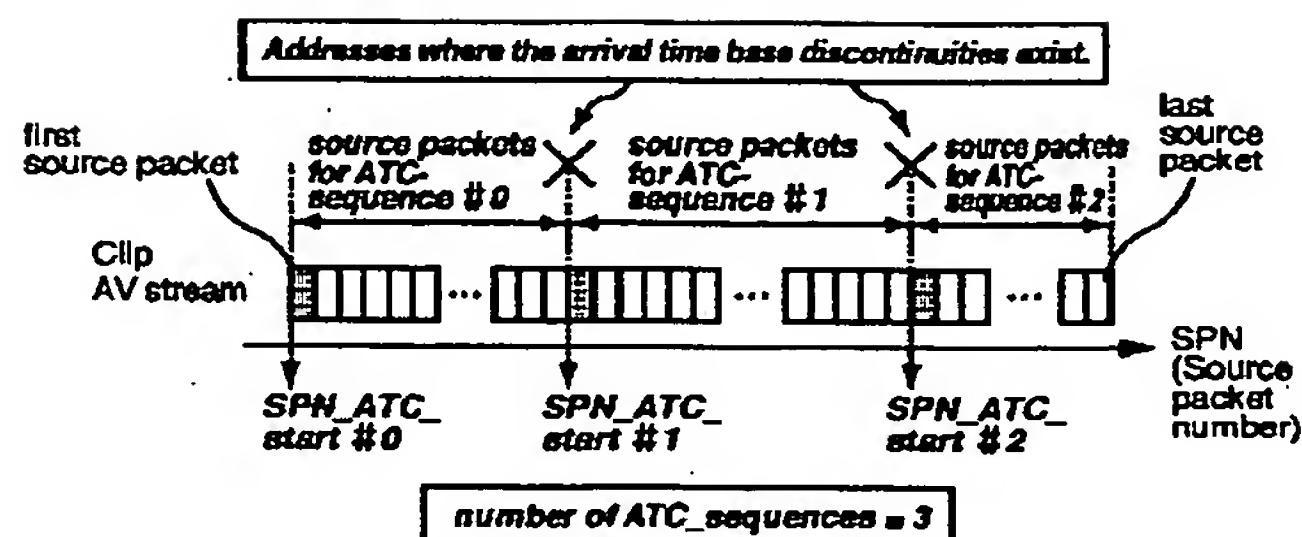
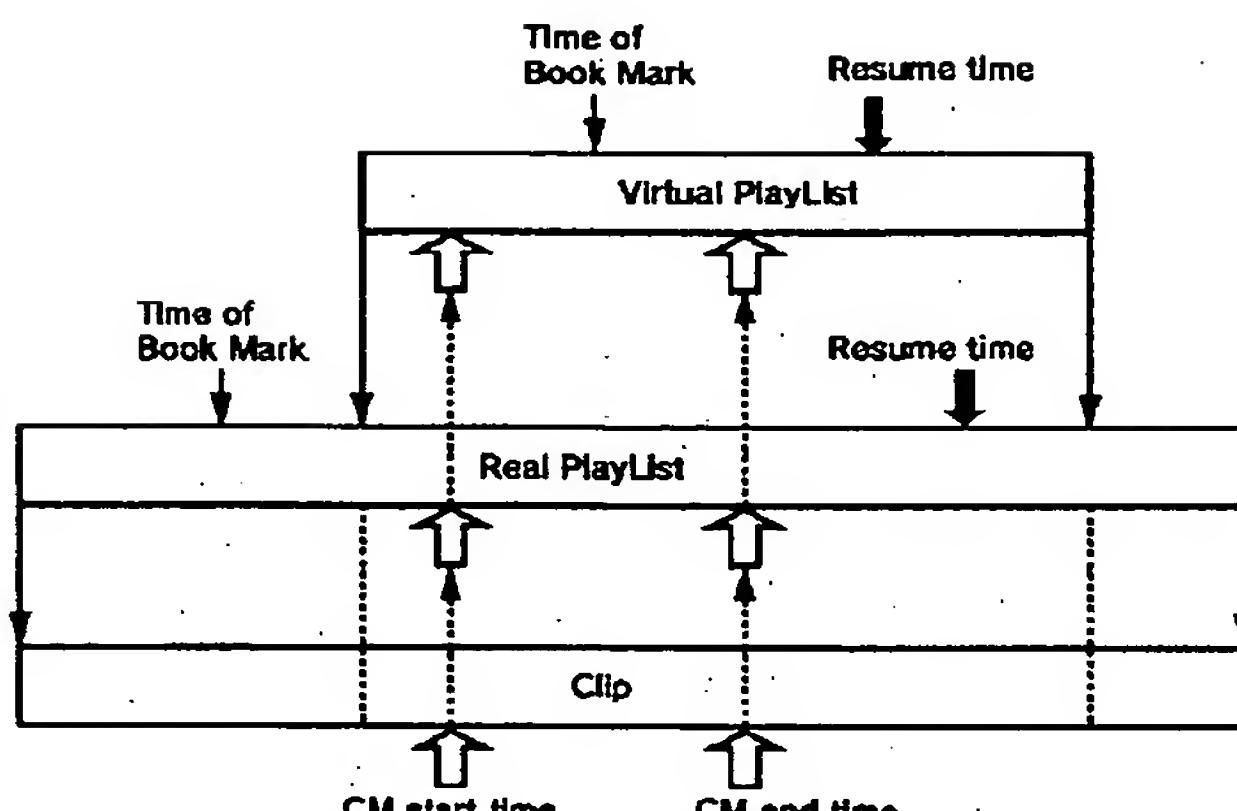


図10 ATCの不連続点とATC-sequence の関係を説明する図

【図27】



PlayListMark と ClipMark の関係について説明する図

図27

【図11】

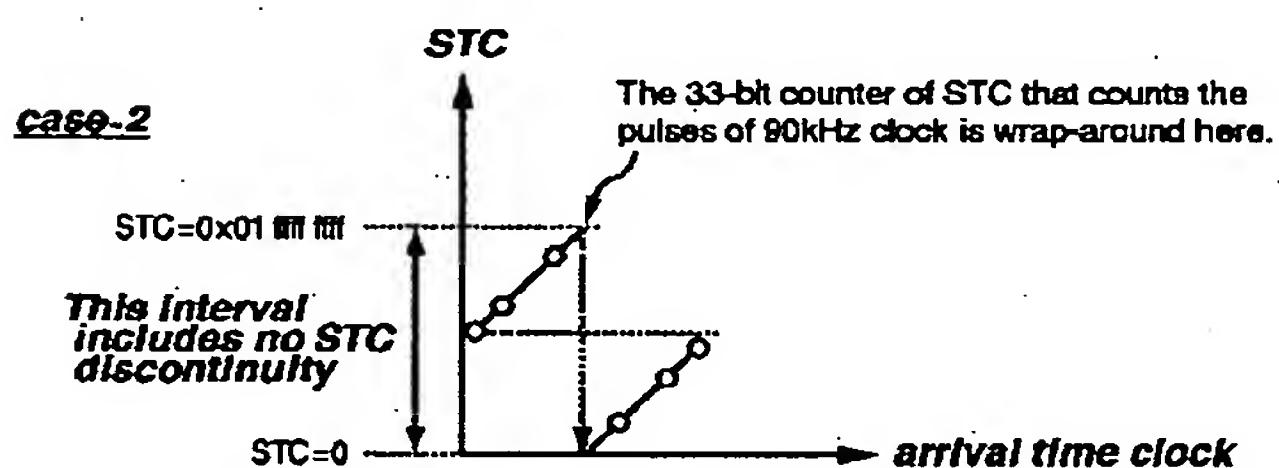
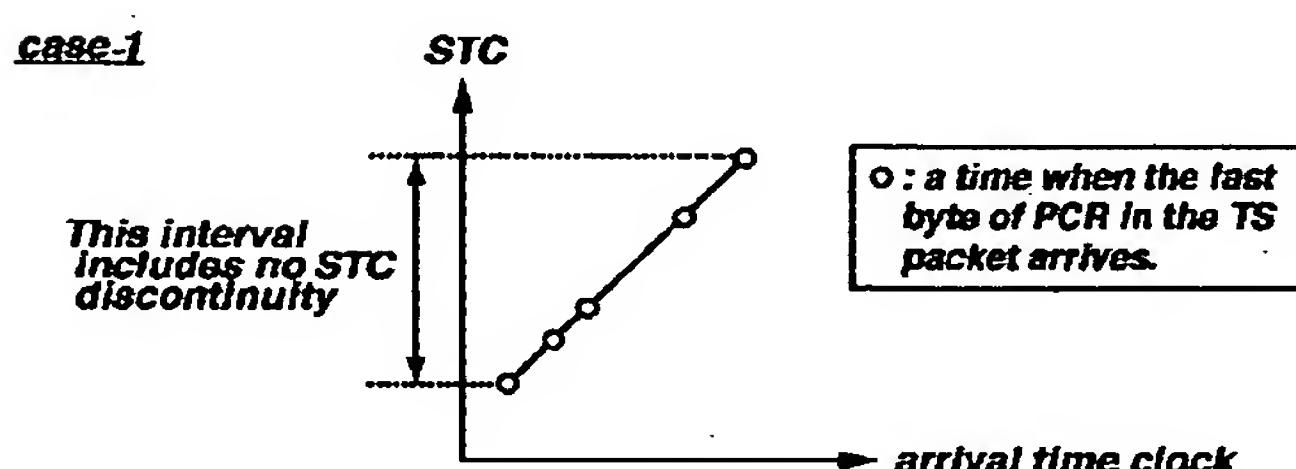


図11 連続なSTC区間について説明する図

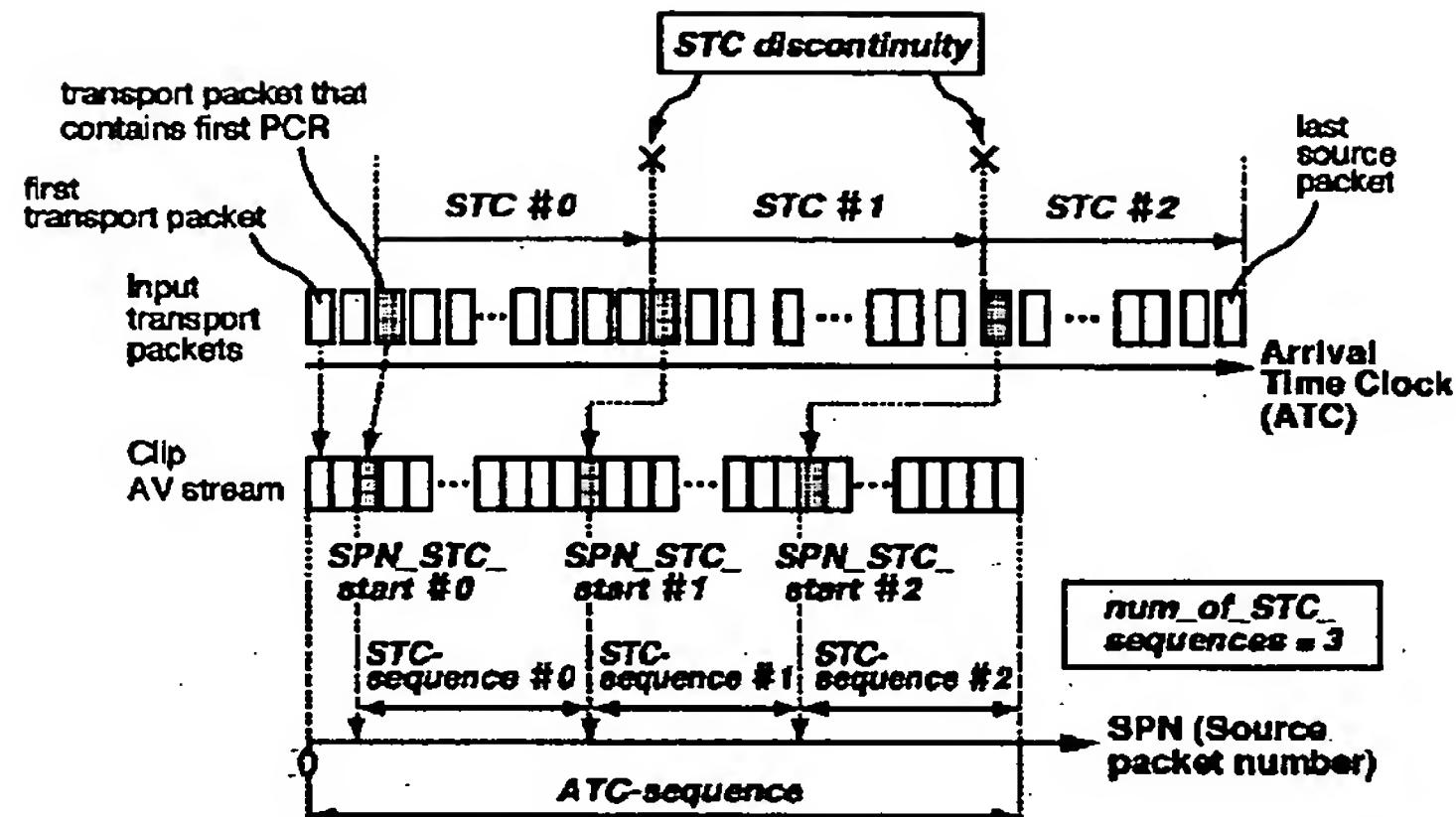
【図18】

video_format

| video_format | Meaning | Video_standard |
|--------------|-------------------------|----------------|
| 0 | 480I | ITU-R BT.601-4 |
| 1 | 576i | ITU-R BT.601-4 |
| 2 | 480p | SMPTE 293M |
| 3 | 1080I | SMPTE 274M |
| 4 | 720p | SMPTE 296M |
| 5-14 | reserved for future use | |
| 15 | No information | |

図18

【図12】



STCの不連続点とSTC-sequenceの関係、およびSTC-sequenceとATC-sequenceの関係を説明する図

図12

【図13】

| Syntax | No. of bits | Mnemonic |
|--|-------------|----------|
| Sequenceinfo() { | | |
| length | 32 | uimsbf |
| reserved_for_word_align | 8 | bsbf |
| num_of_ATC_sequences | 8 | uimsbf |
| for(atc_id=0; atc_id<num_of_ATC_sequences; atc_id++) { | | |
| SPN_ATC_start[atc_id] | 32 | uimsbf |
| num_of_STC_sequences[atc_id] | 8 | uimsbf |
| offset_STC_id[atc_id] | 8 | uimsbf |
| for (stc_id = offset_STC_id[atc_id]; | | |
| stc_id | | |
| <num_of_STC_sequences[atc_id]+offset_STC_id[atc_id]; | | |
| stc_id++) { | | |
| PCR_PID[atc_id][stc_id] | 16 | uimsbf |
| SPN_STC_start[atc_id][stc_id] | 32 | uimsbf |
| presentation_start_time[atc_id][stc_id] | 32 | uimsbf |
| presentation_end_time[atc_id][stc_id] | 32 | uimsbf |
| } | | |
| } | | |

SequenceInfo() のシンタクス

図13

【図20】

display_aspect_ratio

| display_aspect_ratio | Meaning |
|----------------------|-----------------------------|
| 0 | reserved for future use |
| 1 | reserved for future use |
| 2 | 4:3 display aspect ratio |
| 3 | 16:9 display aspect ratio |
| 4 | 2.21:1 display aspect ratio |
| 5-14 | reserved for future use |
| 15 | No information |

図20

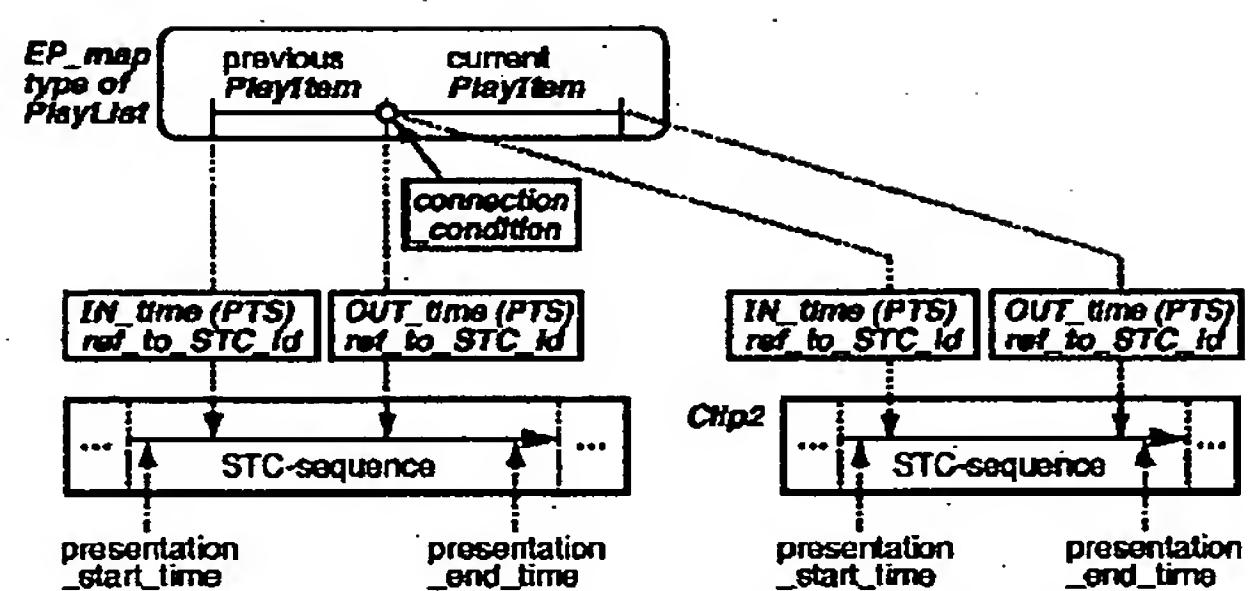


図32 EP-map type の Playlist について説明する図

【図15】

| Syntax | No. of bits | Mnemonic |
|--|-------------|----------|
| programInfo() { | | |
| length | 32 | uimsbf |
| reserved_for_word_align | 8 | bslbf |
| num_of_program_sequences | 8 | uimsbf |
| for(i=0; i<num_of_program_sequences; i++){ | | |
| SPN_program_sequences_start | 32 | uimsbf |
| program_map_PID | 16 | bslbf |
| num_of_streams_in_ps | 8 | uimsbf |
| num_of_groups | 8 | uimsbf |
| for (stream_index=0; stream_index<num_of_streams_in_ps; stream_index++){ | | |
| stream_PID | 16 | uimsbf |
| StreamCodingInfo() | | |
| } | | |
| if (num_of_groups>1){ | | |
| for(l=0; l<num_of_groups; l++){ | | |
| num_of_streams_in_group | 8 | uimsbf |
| for (k=0; k<num_of_streams_in_group; k++){ | | |
| stream_index | 8 | uimsbf |
| } | | |
| if (num_of_streams_in_group%2==0){ | | |
| reserved_for_word_align | 8 | bslbf |
| } | | |
| } | | |
| } | | |
| } | | |

ProgramInfo() のシンタクス

図15

【図17】

stream_coding_type

| stream_coding_type | Meaning |
|--------------------|---|
| 0x00 - 0x01 | reserved for future use |
| 0x02 | MPEG-1 or MPEG-2 video stream |
| 0x03 | MPEG-1 audio |
| 0x04 | MPEG-2 multi-channel audio, backward compatible to MPEG-1 |
| 0x05 | reserved for future use |
| 0x06 | Teletext defined in SESF or DVB or Subtitle defined in ISDB |
| 0x07 - 0x09 | reserved for future use |
| 0x0A | ISO/IEC 13818-6 type A |
| 0x0B | ISO/IEC 13818-6 type B |
| 0x0C | ISO/IEC 13818-6 type C |
| 0x0D | ISO/IEC 13818-6 type D |
| 0x0E | reserved for future use |
| 0x0F | MPEG-2 AAC audio with ADTS transport syntax |
| 0x10 - 0x7F | reserved for future use |
| 0x80 | SESF LPCM audio |
| 0x81 | Dolby AC-3 audio |
| 0x82 - 0xFF | reserved for future use |

図17

【図16】

| Syntax | No. of bit | Mnemonic |
|---|------------|----------|
| StreamCodingInfo() | | |
| length | 8 | bslbf |
| stream_coding_type | 8 | ulmsbf |
| if (stream_coding_type==0x02){ | | |
| video_format | 4 | ulmsbf |
| frame_rate | 4 | ulmsbf |
| display_aspect_ratio | 4 | ulmsbf |
| reserved_for_word_align | 2 | bslbf |
| cc_flag | 1 | ulmsbf |
| original_video_format_flag | 1 | |
| if (original_video_format_flag==1){ | | |
| original_video_format | 4 | ulmsbf |
| original_display_aspect_ratio | 4 | ulmsbf |
| reserved_for_word_align | 8 | bslbf |
| } | | |
| } else if (stream_coding_type==0x03// stream_coding_type==0x04// stream_coding_type==0x0F// stream_coding_type==0x80// stream_coding_type==0x81){ | | |
| audio_presentation_type | 4 | ulmsbf |
| sampling_frequency | 4 | ulmsbf |
| reserved_for_word_align | 8 | bslbf |
| } | | |
| } | | |

StreamCodingInfo() のシンタクス

図16

【図19】

frame_rate

| frame_rate | Meaning |
|------------|-------------------------|
| 0 | reserved for future use |
| 1 | 24 000/1001 (23.976...) |
| 2 | 24 |
| 3 | 25 |
| 4 | 30 000/1001 (29.97...) |
| 5 | 30 |
| 6 | 50 |
| 7 | 60 000/1001 (59.94...) |
| 8 | 60 |
| 9-14 | reserved for future use |
| 15 | No information |

図19

【図22】

sampling_frequency

| sampling_frequency | Meaning |
|--------------------|-------------------------|
| 0 | 48 kHz |
| 1 | 44.1 kHz |
| 2 | 32 kHz |
| 3-14 | reserved for future use |
| 15 | No information |

図22

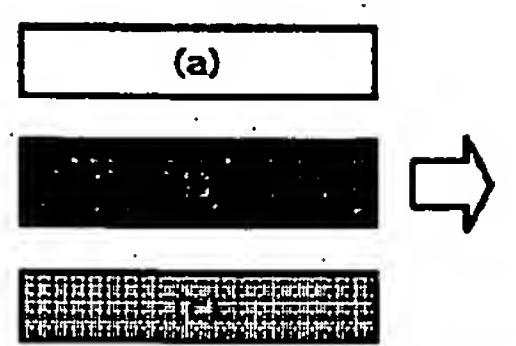
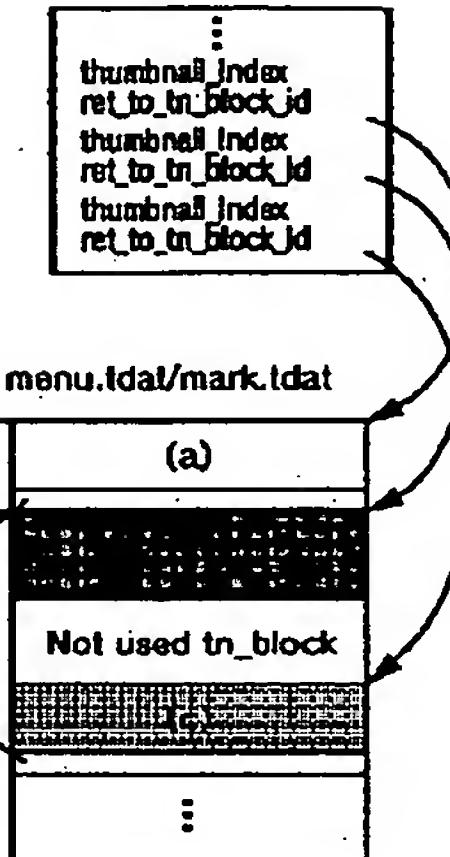
【図21】

audio_presentation_type

| audio_presentation_type | Meaning |
|-------------------------|---|
| 0 | reserved for future use |
| 1 | single mono channel |
| 2 | dual mono channel |
| 3 | stereo (2-channel) |
| 4 | multi-lingual |
| 5 | surround sound |
| 6 | multi-channel |
| 7-12 | reserved for future use |
| 13 | audio description for the visually impaired |
| 14 | audio for the hard of hearing |
| 15 | No information |

図21

【図43】

picture data**menu.tidx/mark.tidx**

tn_blockへのデータの格納方法を説明する図

【図23】

図43

| Syntax | No. of bits | Mnemonic |
|-------------------------|-------------|----------|
| CPI(){ | | |
| length | 32 | uimsbf |
| reserved_for_word_align | 15 | bslbf |
| CPI_type | 1 | bslbf |
| if (CPI_type == 0){ | | |
| EP_map() | | |
| } else { | | |
| TU_map() | | |
| } | | |
| } | | |

CPI() のシンタクス

図23

【図25】

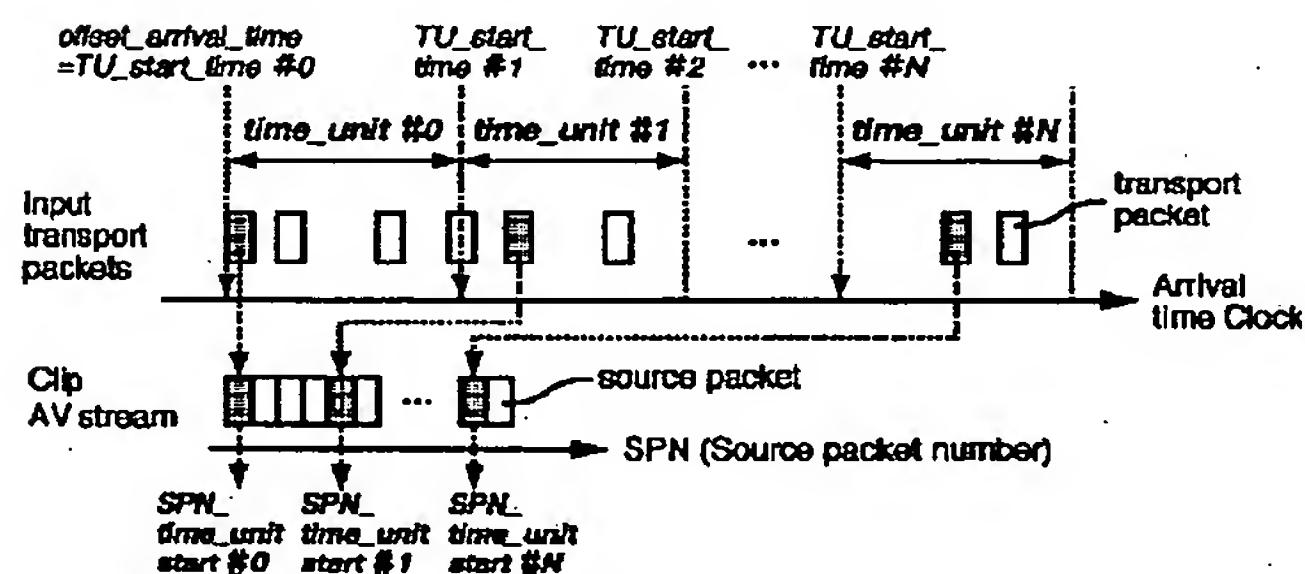


図25 AVストリームをはじめてClipとして記録するときにできる TU-mapについて説明する図

【図26】

| Syntax | No. of bits | Mnemonic |
|--|-------------|----------|
| TU_map() | | |
| time_unit_size | 32 | uimsbf |
| for(atc_id=0; atc_id<num_of_ATC_sequences; atc_id++) { | | |
| offset_arrival_time[atc_id] | 32 | bslbf |
| num_of_time_unit_entries[atc_id] | 32 | uimsbf |
| } | | |
| for(atc_id=0; atc_id<num_of_ATC_sequences; atc_id++) { | | |
| for(i=0; i<num_of_time_unit_entries[atc_id]; i++) { | | |
| SPN_time_unit_start[atc_id][i] | 32 | uimsbf |
| } | | |
| } | | |
| } | | |

図26 TU_map のシンタクス

【図28】

| Syntax | No. of bits | Mnemonic |
|---|-------------|----------|
| ClipMark() { | | |
| length | 32 | uimsbf |
| marker_ID | 16 | uimsbf |
| number_of_Clip_marks | 16 | uimsbf |
| for(i=0; i < number_of_Clip_marks; i++) { | | |
| mark_invalid_flag | 1 | uimsbf |
| mark_type | 7 | uimsbf |
| ref_to_STC_id | 8 | uimsbf |
| mark_time_stamp | 32 | uimsbf |
| entry_ES_PID | 16 | uimsbf |
| ref_to_thumbnail_index | 16 | uimsbf |
| representative_picture_time_stamp | 32 | uimsbf |
| } | | |
| } | | |

ClipMark のシンタクス

図28

【図33】

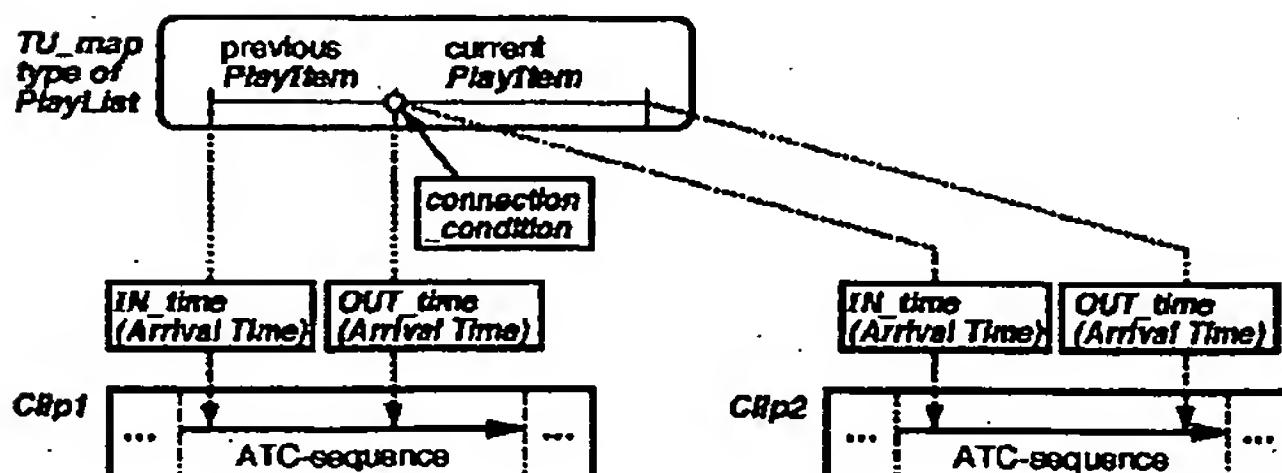


図33 TU-map type の Playlist について説明する図

【図45】

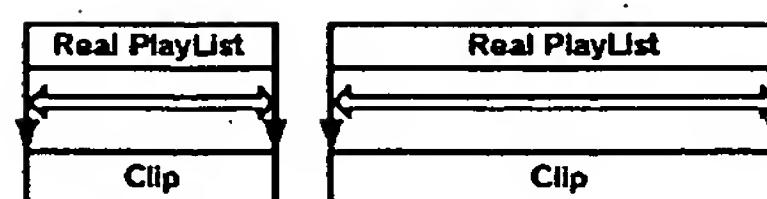


図45 AVストリームが新しいClipとして記録される時のClipと Playlist の関係のコンセプトを説明する図

【図46】

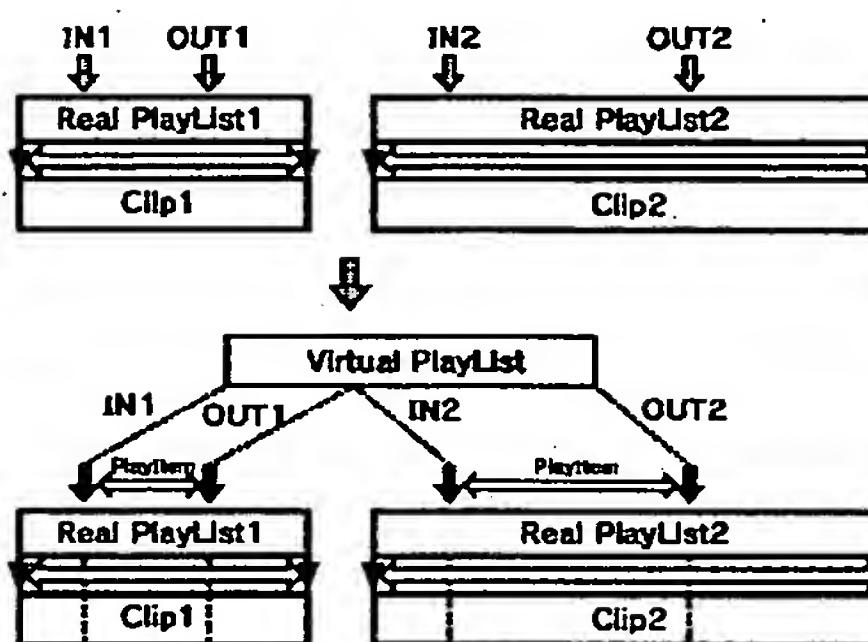


図46 Virtual Playlist の作成のコンセプトについて説明する図

【図47】

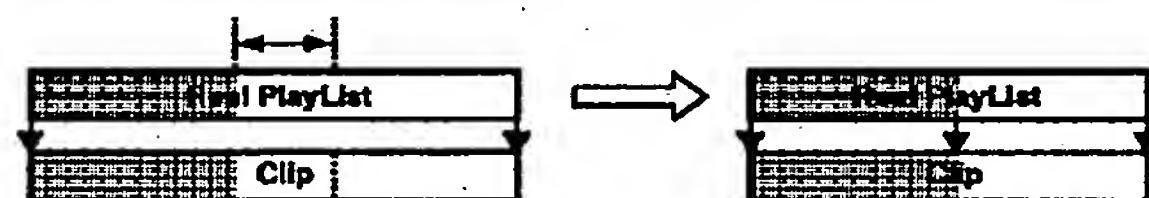


図47 Real Playlist の再生区間の一部分を消去したときの Clip と Playlist の関係のコンセプトを説明する図

【図29】

| Syntax | No. of bits | Mnemonic |
|---------------------------------|-------------|----------|
| xxxxx.rpls / yyyy.vpls { | | |
| version_number | 8*4 | bslbf |
| PlayList_start_address | 32 | uimsbf |
| PlayListMark_start_address | 32 | uimsbf |
| MakersPrivateData_start_address | 32 | uimsbf |
| reserved_for_future_use | 160 | bslbf |
| UIAppInfoPlayList() | | |
| for(i=0; i<N1; i++) { | | |
| padding_word | 16 | bslbf |
| } | | |
| PlayList() | | |
| for(i=0; i<N2; i++) { | | |
| padding_word | 16 | bslbf |
| } | | |
| PlayListMark() | | |
| for(i=0; i<N3; i++) { | | |
| padding_word | 16 | bslbf |
| } | | |
| MakersPrivateData() | | |
| for(i=0; i<N4; i++) { | | |
| padding_word | 16 | bslbf |
| } | | |
| } | | |

PlayList file のシンタクス

図29

【図30】

| Syntax | No. of bits | Mnemonic |
|-------------------------|-------------|----------|
| UIAppInfoPlayList() { | | |
| length | 32 | uimsbf |
| reserved_for_future_use | 16 | bslbf |
| PlayList_character_set | 8 | uimsbf |
| reserved_for_word_align | 3 | bslbf |
| playback_control_flag | 1 | uimsbf |
| write_protect_flag | 1 | uimsbf |
| is_played_flag | 1 | uimsbf |
| archive | 2 | uimsbf |
| record_time_and_date | 4*14 | bslbf |
| PlayList_duration | 4*6 | bslbf |
| maker_ID | 16 | uimsbf |
| maker_model_code | 16 | uimsbf |
| ref_to_thumbnail_index | 16 | uimsbf |
| channel_number | 16 | uimsbf |
| reserved_for_word_align | 8 | bslbf |
| channel_name_length | 8 | uimsbf |
| channel_name | 8*20 | bslbf |
| PlayList_name_length | 8 | uimsbf |
| PlayList_name | 8*255 | bslbf |
| PlayList_detail_length | 16 | uimsbf |
| PlayList_detail | 8*1200 | bslbf |
| } | | |

UIAppInfoPlayList のシンタクス

図30

【図49】

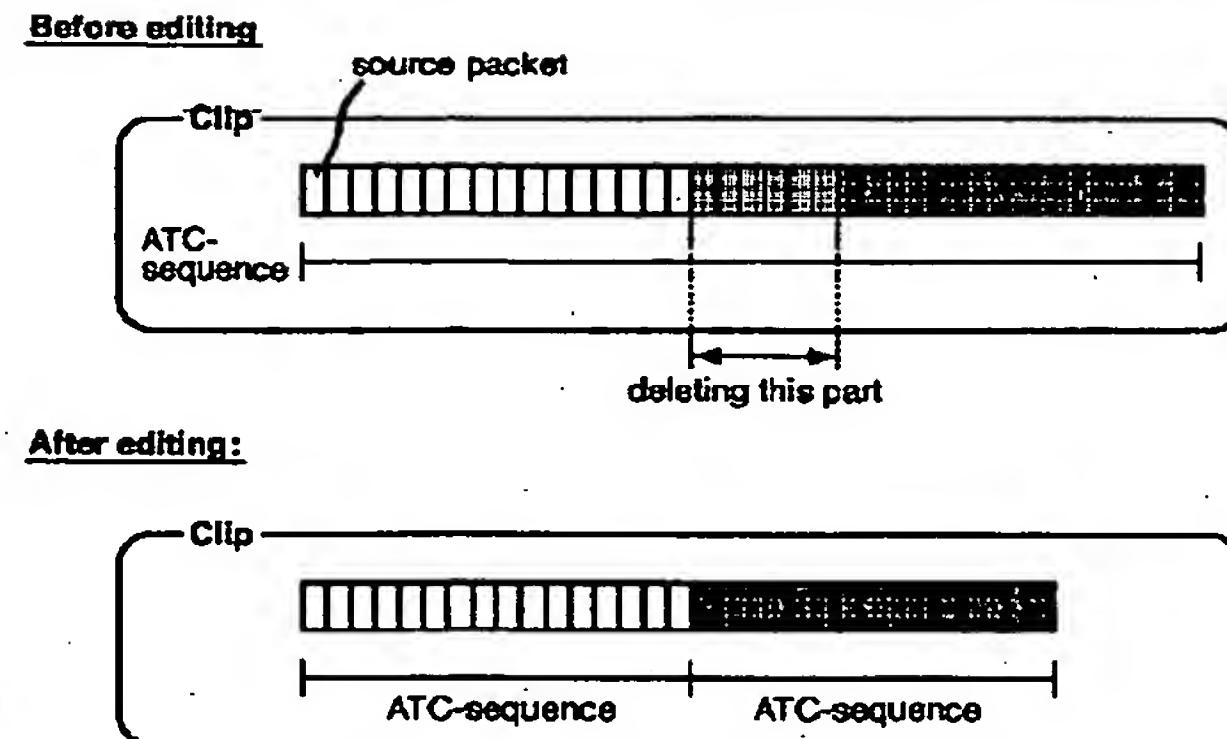
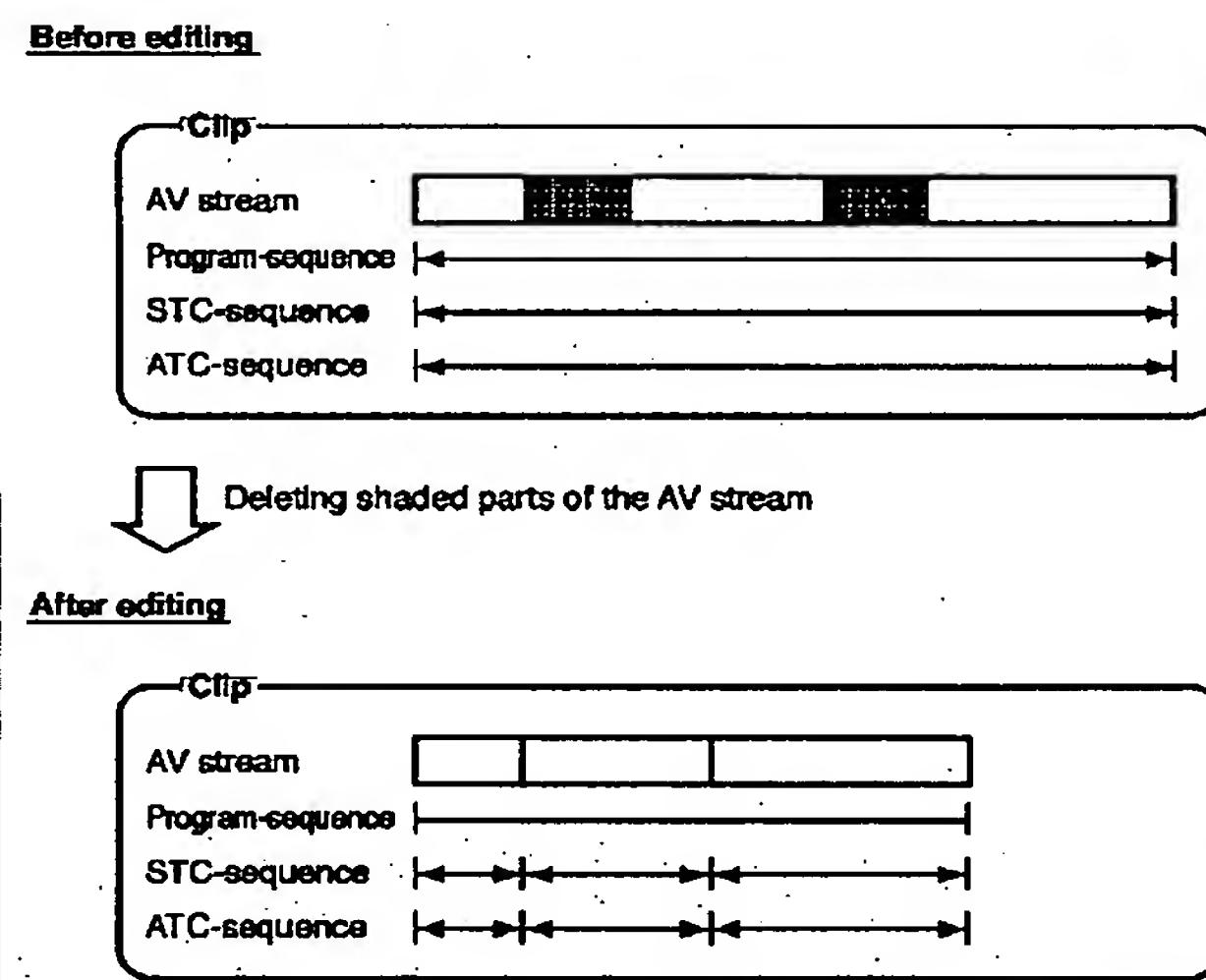


図49 1つの Clip AVストリームのデータを部分的に消去した時に、Clipの中に2つのATC-sequencesができる場合を説明する図

【図50】



1つの Clip AVストリームのデータを部分的に消去した時の ATC-sequence、STC-sequence および program-sequence の関係を説明する図

図50

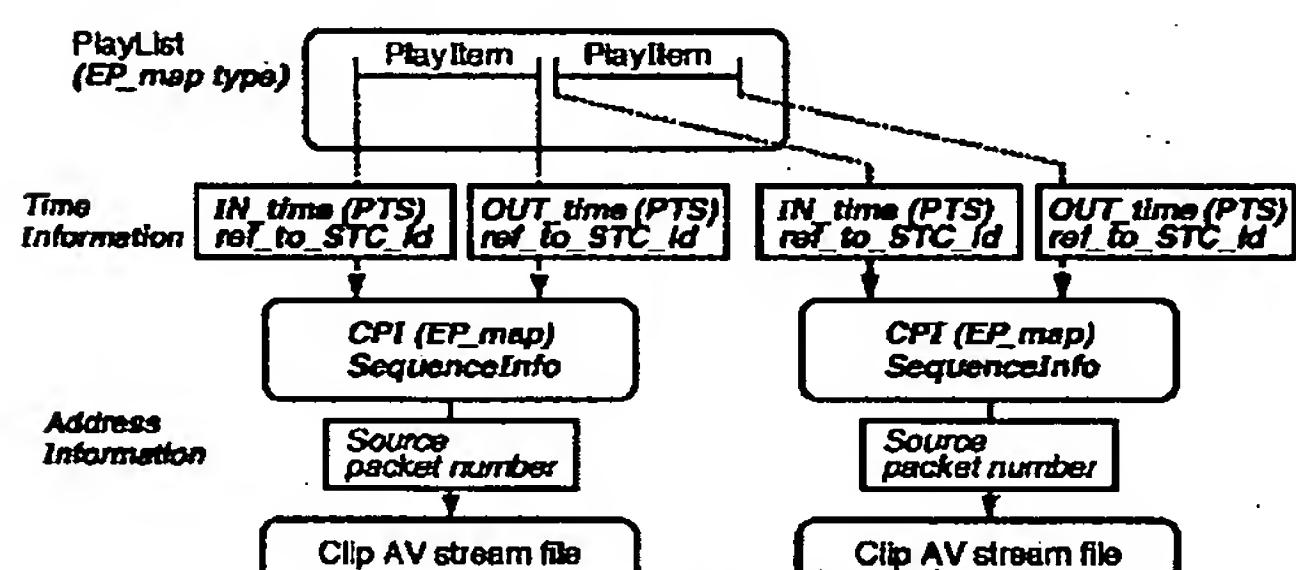
【図31】

| Syntax | No. of bits | Mnemonic |
|---|-------------|----------|
| PlayList() | | |
| length | 32 | ulmsbf |
| reserved_for_word_align | 15 | bslbf |
| CPI_type | 1 | bslbf |
| number_of_PlayItems | 16 | ulmsbf |
| if(<Virtual-PlayList> && CPI_type==0) { | | |
| number_of_SubPlayItems | 16 | ulmsbf |
| } else { | | |
| reserved_for_word_align | 16 | bslbf |
| } | | |
| for (PlayItem_id=0; PlayItem_id<number_of_PlayItems; PlayItem_id++) { | | |
| PlayItem() | | |
| } | | |
| if(<Virtual-PlayList> && CPI_type==0) { | | |
| for (l=0; l<number_of_SubPlayItems; l++) { | | |
| SubPlayItem() | | |
| } | | |
| } | | |
| } | | |

PlayList() のシンタクス

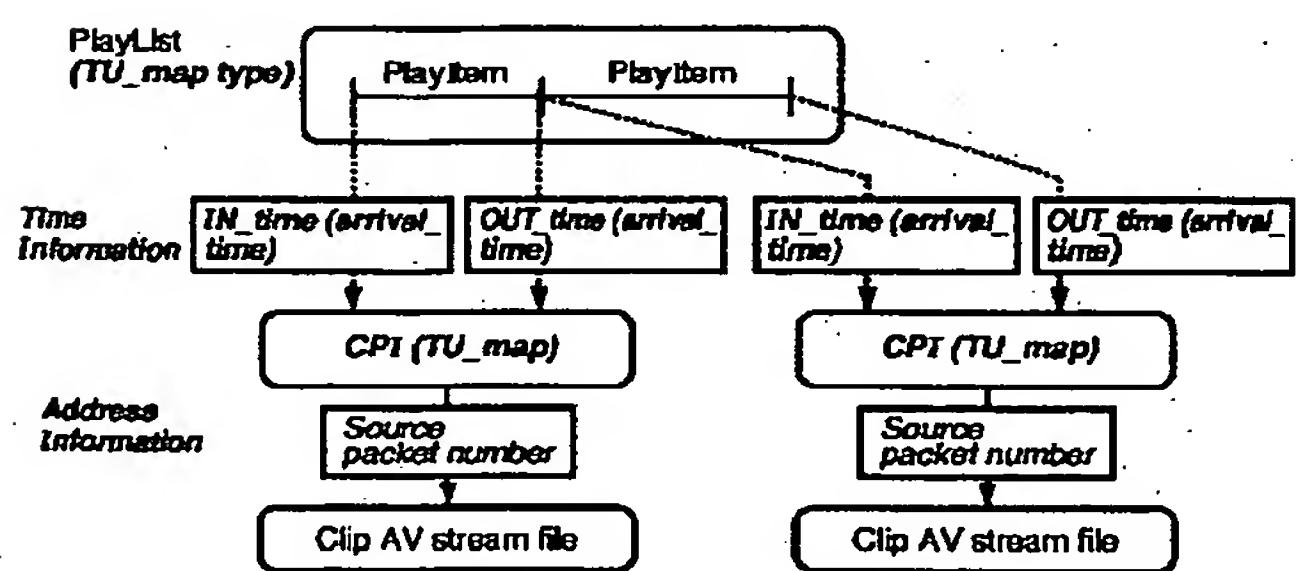
図31

【図34】



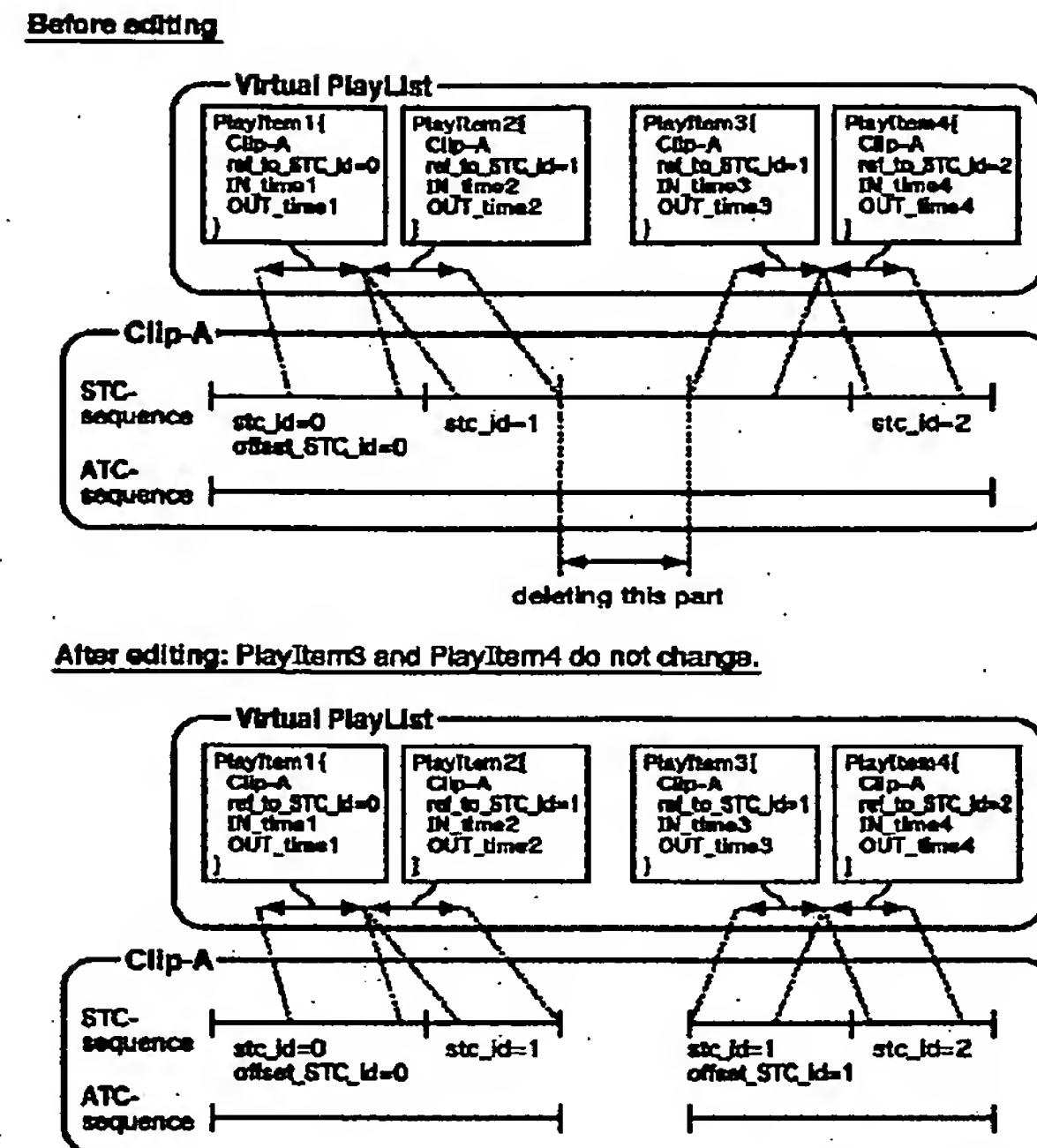
EP-map type の PlayList の時間情報とAVストリームファイルの中のアドレス情報の関係を説明する図

【図35】



TU-map type の PlayList の時間情報とAVストリームファイルの中のアドレス情報の関係を説明する図

【図51】



CPIがEP_mapである Clip AVストリームの一部分を消去した時の Clip と PlayList の関係を説明する図

図51

【図52】

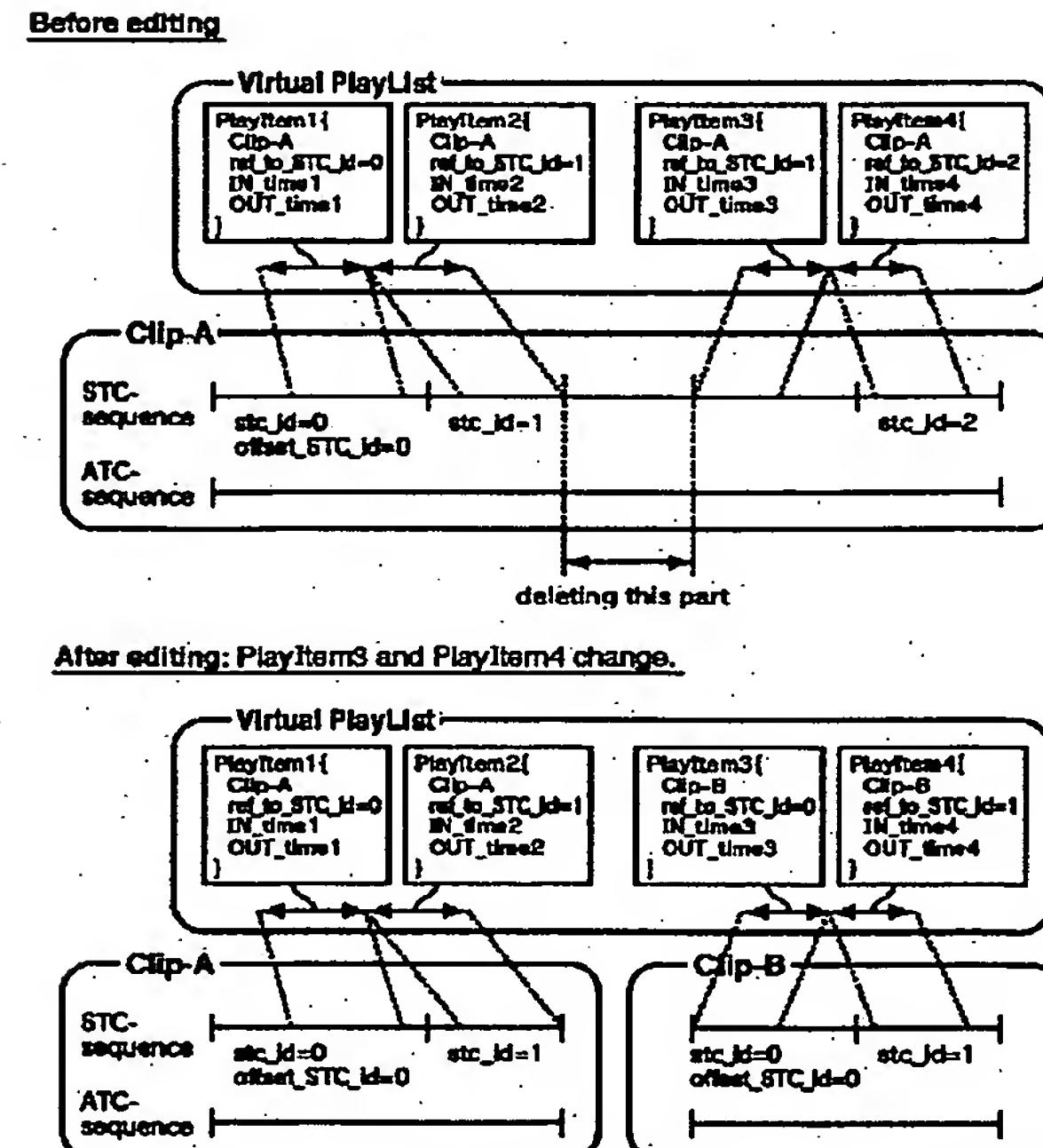


図52

【図36】

| Syntax | No. of bits | Mnemonic |
|---|-------------|----------|
| PlayItem(){ | | |
| length | 16 | uimsbf |
| Clip_Information_file_name | 8*10 | bslbf |
| reserved_for_word_align | 6 | bslbf |
| connection_condition | 2 | bslbf |
| if(CPI_type==0){ /* the CPI_type is defined in the Playlist(). */ | | |
| ref_to_STC_Id | 8 | uimsbf |
| } else { | | |
| reserved_for_word_align | 8 | bslbf |
| } | | |
| IN_time | 32 | uimsbf |
| OUT_time | 32 | uimsbf |
| if(<Virtual-Playlist> && connection_condition=='10') { | | |
| Bridge_Clip_Information_file_name | | |
| } | | |
| } | | |

PlayItem() のシンタクス

図36

【図37】

| Syntax | No. of bits | Mnemonic |
|--|-------------|----------|
| PlayListMark(){ | | |
| length | 32 | uimsbf |
| number_of_PlayList_marks | 16 | uimsbf |
| for(i=0; i < number_of_PlayList_marks; i++){ | | |
| mark_invalid_flag | 1 | uimsbf |
| mark_type | 7 | uimsbf |
| mark_name_length | 8 | uimsbf |
| ref_to_PlayItem_id | 16 | uimsbf |
| mark_time_stamp | 32 | uimsbf |
| entry_ES_PID | 16 | uimsbf |
| ref_to_thumbnail_index | 16 | uimsbf |
| mark_name | 8*32 | bslbf |
| } | | |
| } | | |

PlayListMark のシンタクス

図37

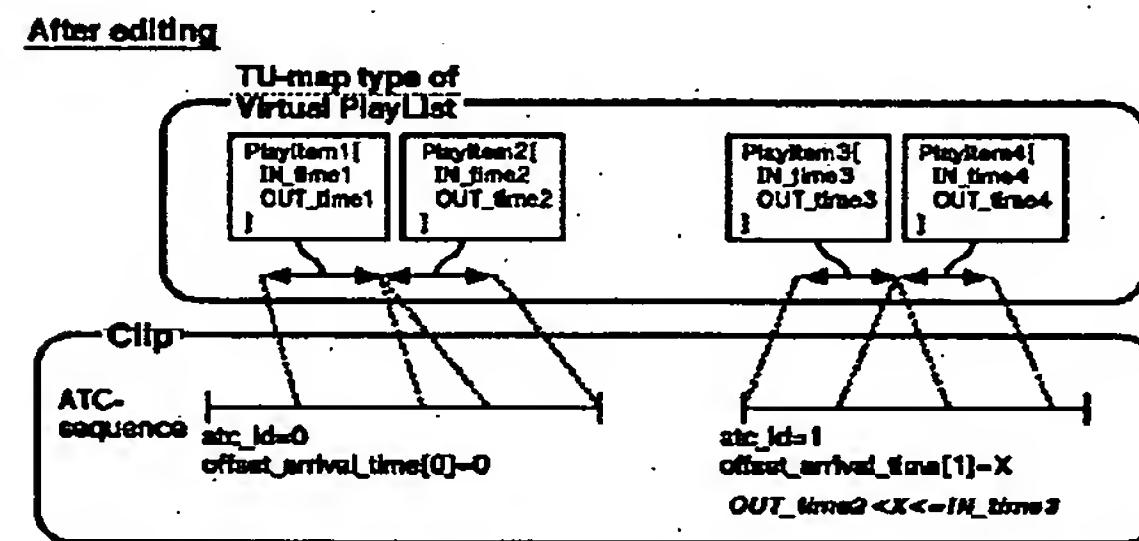
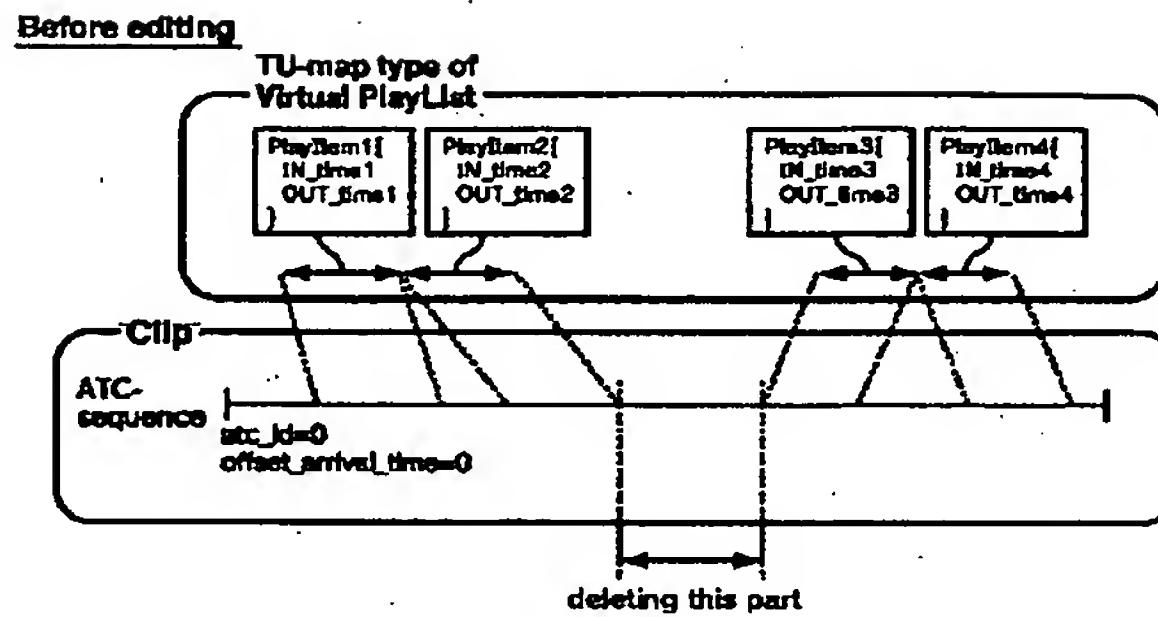
【図39】

| Syntax | No. of bits | Mnemonic |
|---------------------------|-------------|----------|
| UIAppInfoVolume() | | |
| length | 32 | uimsbf |
| reserved_for_future_use | 16 | bslbf |
| volume_character_set | 8 | bslbf |
| reserved_for_word_align | 6 | bslbf |
| volume_protect_flag | 1 | bslbf |
| resume_valid_flag | 1 | bslbf |
| PIN | 8*4 | bslbf |
| resume_PlayList_file_name | 8*10 | bslbf |
| ref_to_thumbnail_index | 16 | uimsbf |
| volume_name_length | 8 | uimsbf |
| volume_name | 8*255 | bslbf |
| } | | |

UIAppInfoVolume のシンタクス

図39

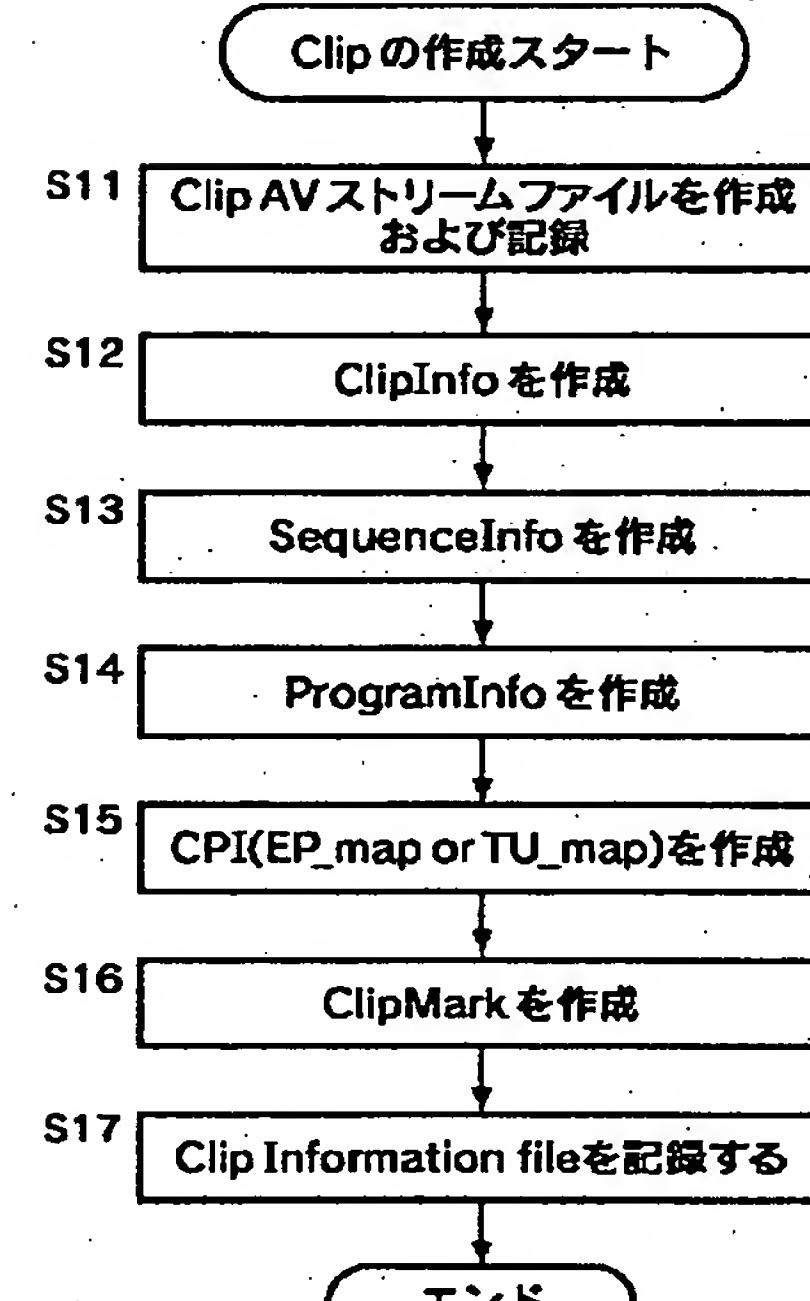
【図53】



CPI が TU_map である Clip AVストリームの一部分を消去した時の Clip と Playlist の関係を説明する図

図53

【図54】



AVストリームを Clip して新しく記録するときの、ClipAVストリームファイルおよび Clip Information ファイルの作成

図54

【図38】

| Syntax | No. of bits | Mnemonic |
|---------------------------------|-------------|----------|
| info.dvr { | | |
| version_number | 8*4 | bslbf |
| TableOfPlayLists_start_address | 32 | uimsbf |
| MakersPrivateData_start_address | 32 | uimsbf |
| reserved_for_future_use | 192 | bslbf |
| UIAppInfoVolume() | | |
| for(i=0; i<N1; i++){ | | |
| padding_word | 16 | bslbf |
| } | | |
| TableOfPlayLists() | | |
| for(i=0; i<N2; i++){ | | |
| padding_word | 16 | bslbf |
| } | | |
| MakersPrivateData() | | |
| for(i=0; i<N3; i++){ | | |
| padding_word | 16 | bslbf |
| } | | |

Info.dvr のシンタクス

図38

【図40】

| Syntax | No. of bits | Mnemonic |
|--|-------------|----------|
| TableOfPlayLists(){ | | |
| length | 32 | uimsbf |
| number_of_PlayLists | 16 | uimsbf |
| for (i=0; i < number_of_PlayLists; i++){ | | |
| PlayList_file_name | 8*10 | bslbf |
| } | | |

TableOfPlayLists のシンタクス 図40

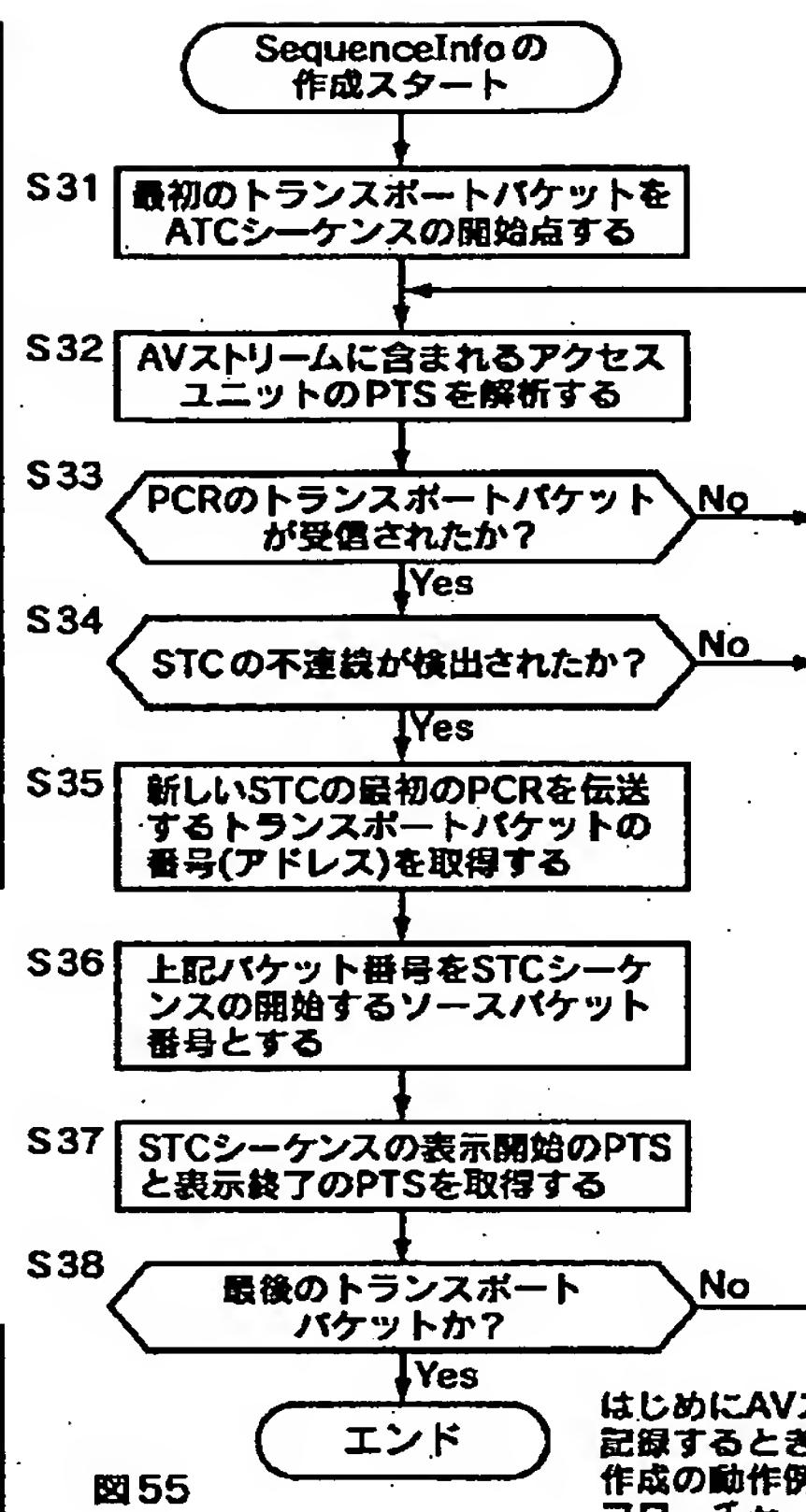
【図42】

| Syntax | No. of bits | Mnemonic |
|--|----------------------|----------|
| menu.tdat / mark.tdat { | | |
| for (tn_block_id=0; tn_block_id<number_of_tn_blocks; tn_block_id++){ | | |
| tn_block | tn_block_size*1024*8 | |
| } | | |

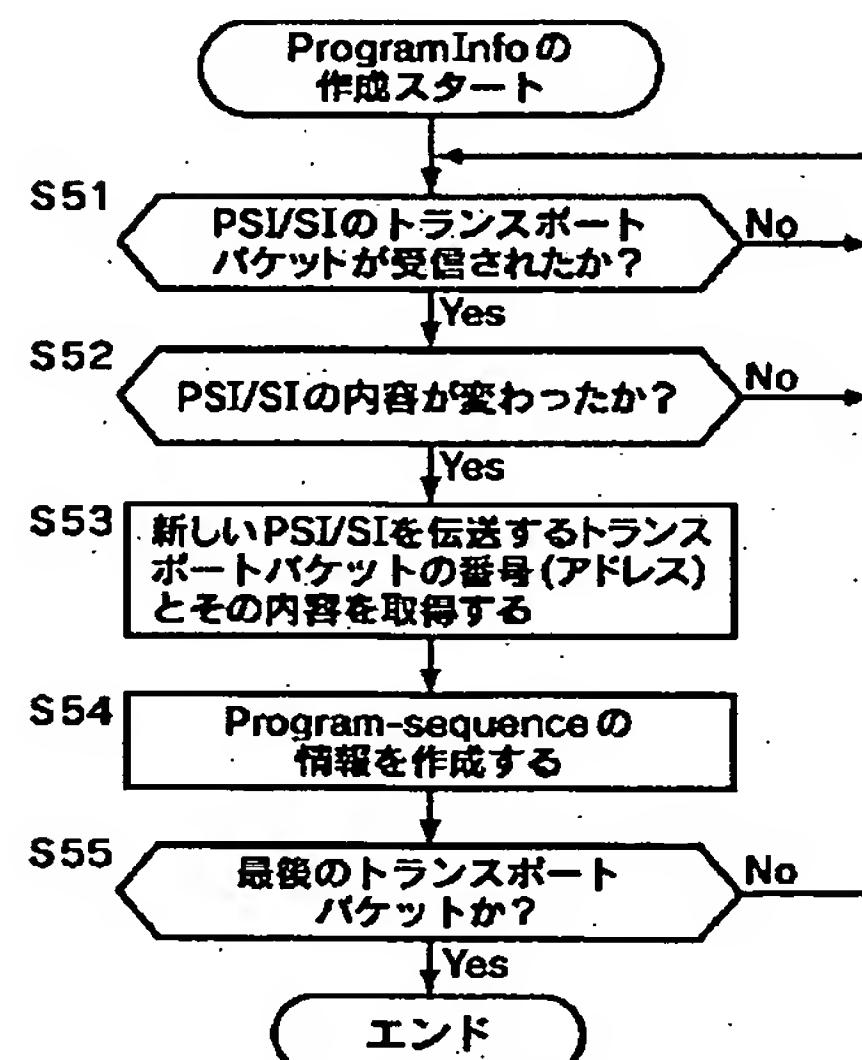
サムネールのピクチャデータファイルのシンタクス

図42

【図55】



【図56】



ProgramInfo の作成の動作例を説明するフローチャート

図56

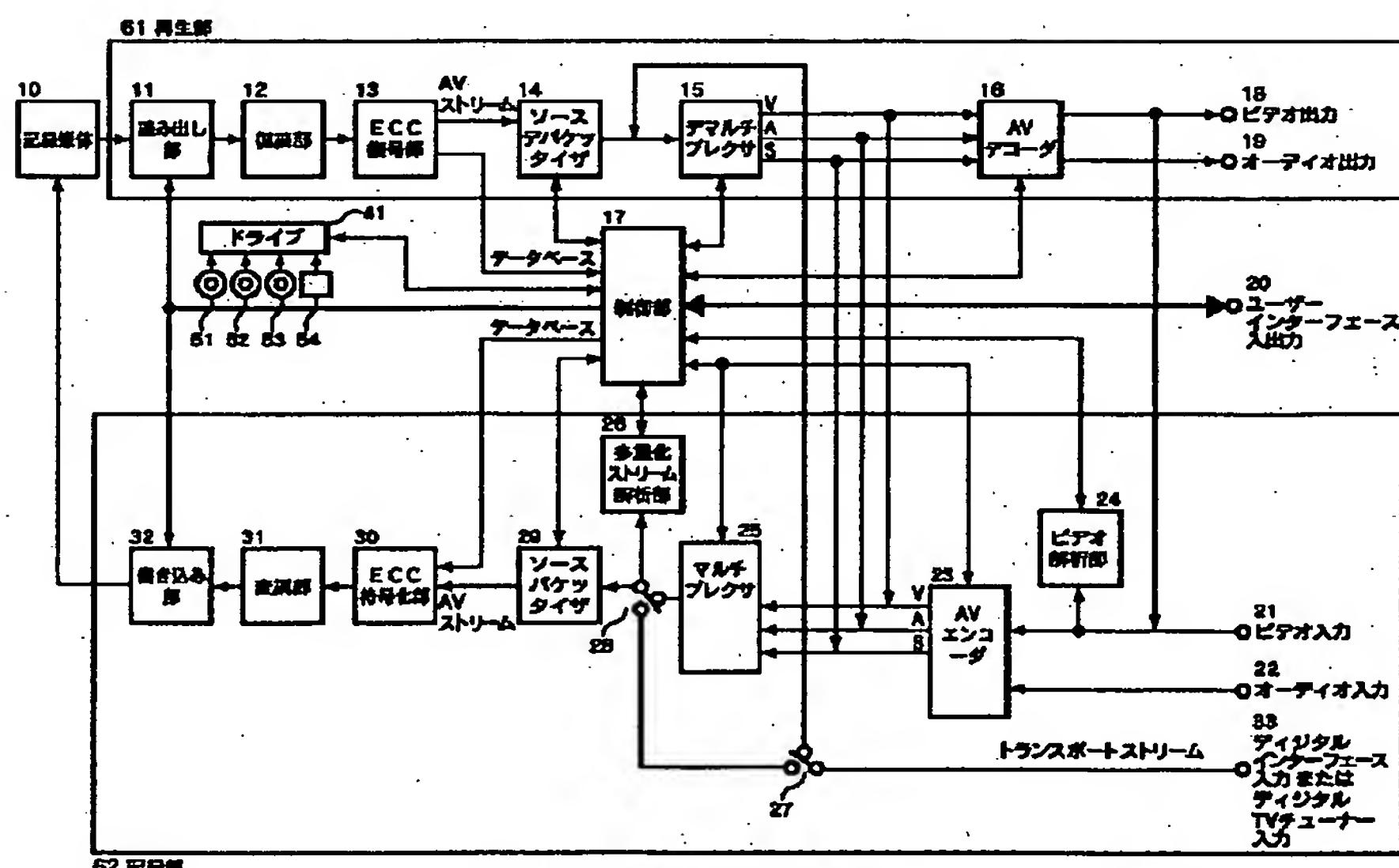
【図41】

| Syntax | No. of bits | Mnemonic |
|---|-------------|----------|
| menu.tidx / mark.tidx { | | |
| version_number | 8*4 | char |
| reserved_for_future_use | 256 | bslbf |
| length | 32 | uimsbf |
| if (length !=0) { | | |
| number_of_thumbnails | 16 | uimsbf |
| tn_block_size | 16 | uimsbf |
| number_of_tn_blocks | 16 | uimsbf |
| for(l=0; l<number_of_thumbnails; l++) { | | |
| thumbnail_index | 16 | uimsbf |
| ref_to_tn_block_id | 16 | uimsbf |
| picture_byte_size | 32 | uimsbf |
| horizontal_picture_size | 16 | uimsbf |
| vertical_picture_size | 16 | uimsbf |
| display_aspect_ratio | 4 | uimsbf |
| color_space | 4 | uimsbf |
| reserved_for_word_align | 8 | bslbf |
| } | | |
| } | | |
| } | | |

サムネールのヘッダ情報ファイルのシンタクス

図41

【図44】



動画像記録再生装置1

図44

【図48】

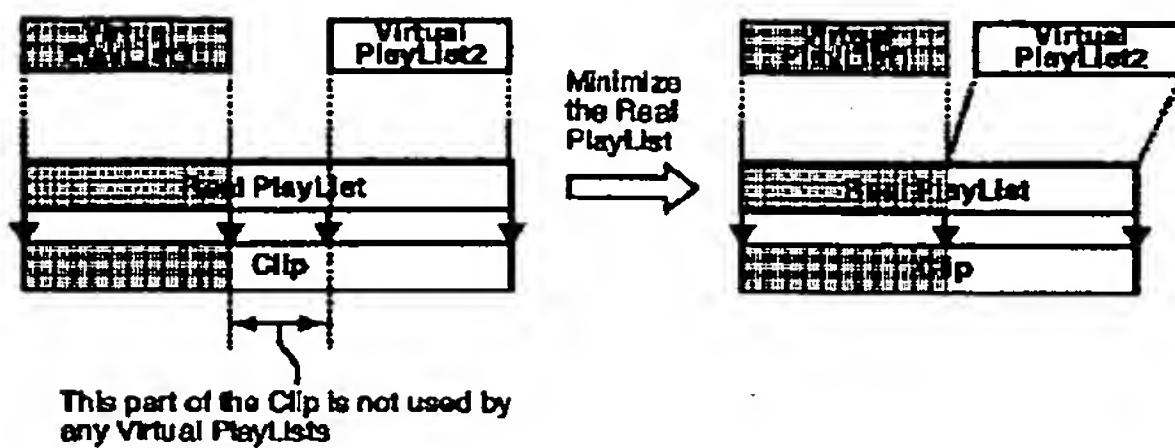


図48 Minimize(ミニマイズ)の編集をしたときの Clip と Real Playlist, Virtual Playlist の関係のコンセプトを説明する図

【図58】



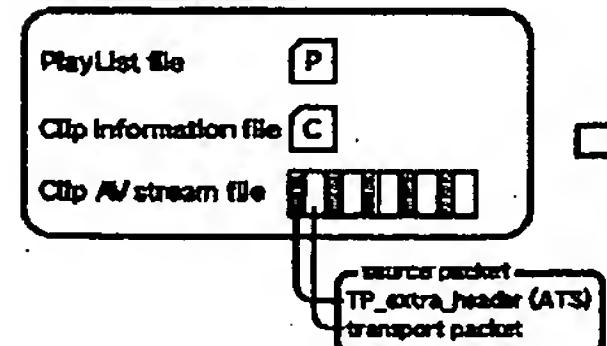
Real Playlist の作成方法を説明するフローチャート

図58

【図61】

AVストリームファイルとデータベースをファイル転送する場合

コピー元の記録媒体

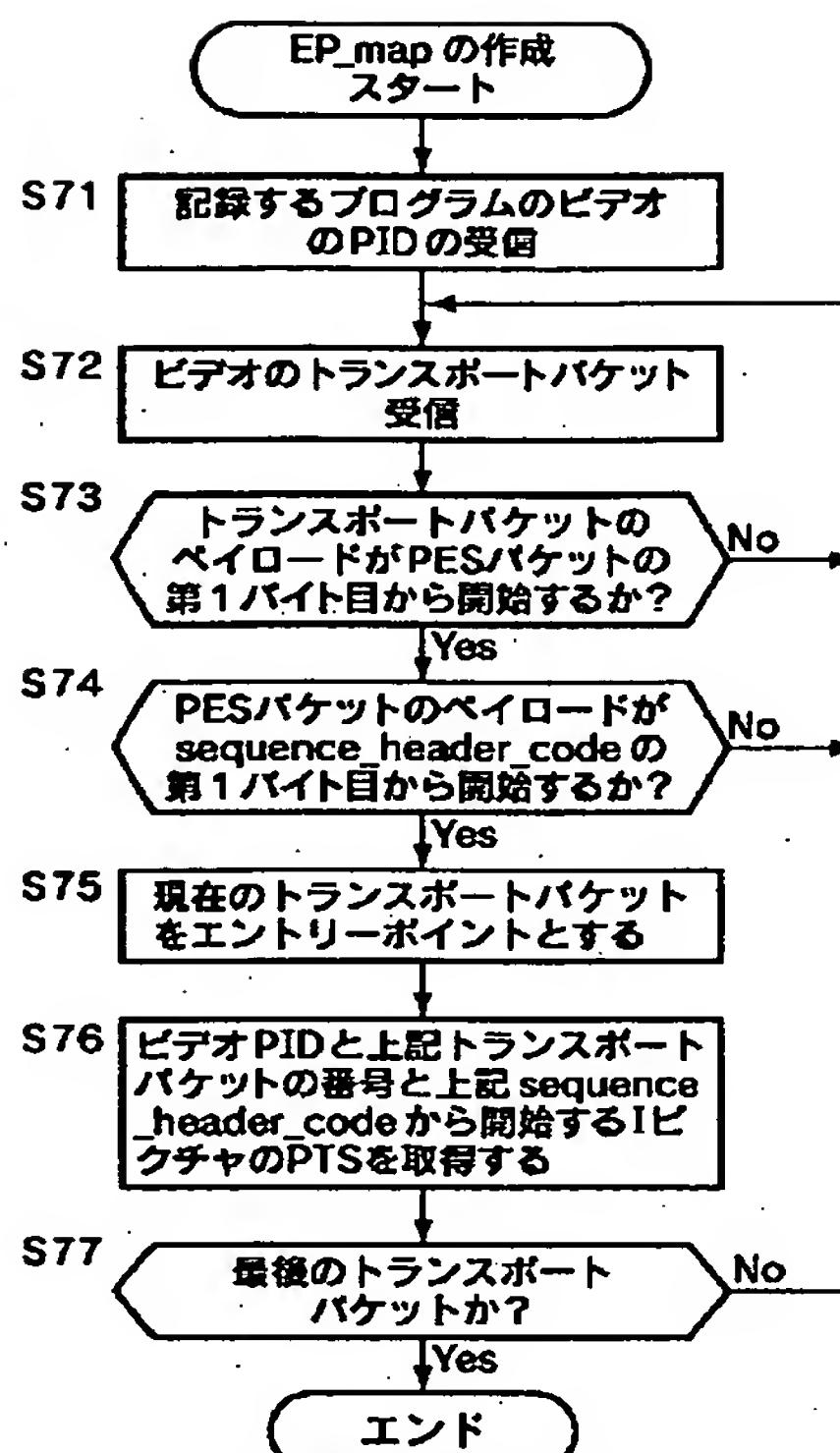


・コピーされた各ソースパケットの
TP_extra_header(ATS)の内容は、コピー元と同じ。
・コピーされたAVストリームに対応するデータベース
ファイルもコピーされる。

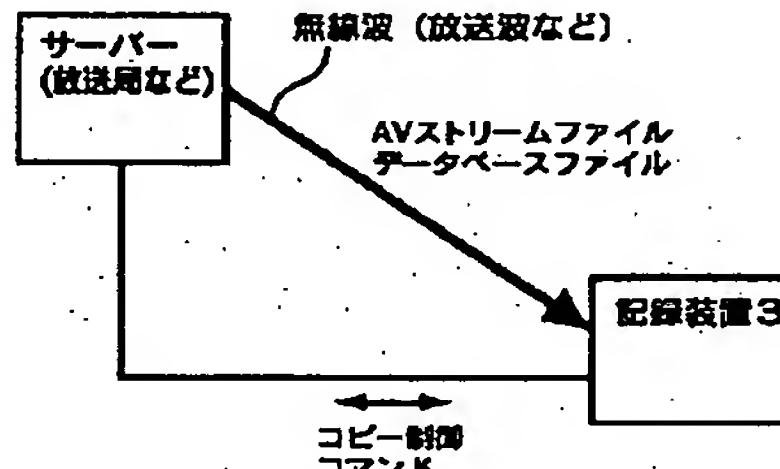
AVストリームファイルとデータベースファイルを共にファイル転送する場合を説明する図

図61

【図57】

EP_map の作成の動作例を説明するフローチャート
図57

【図75】



AVストリームファイルとデータベースファイルが、無線波を経由して、記録装置へファイルが転送される場合

図75

【図59】

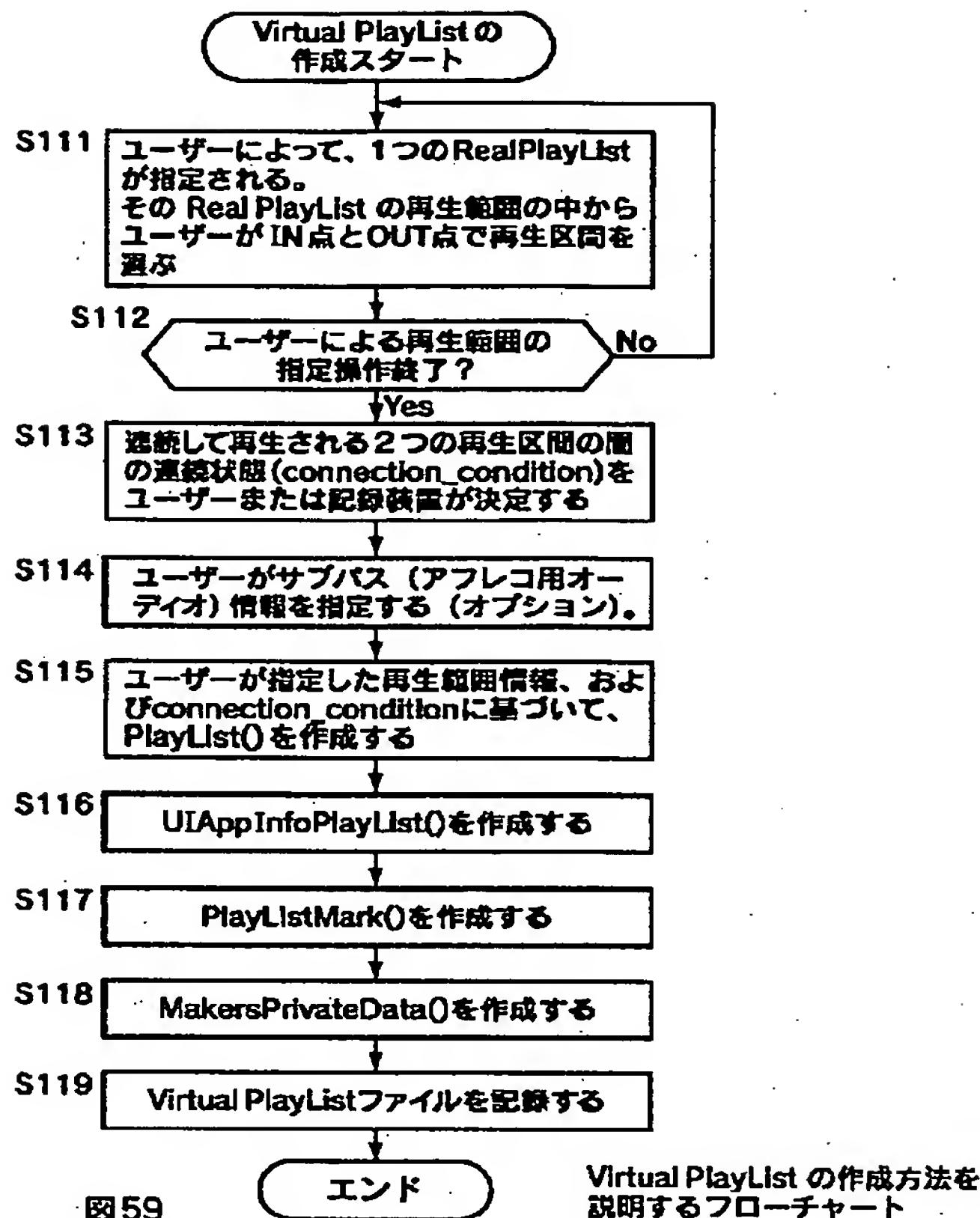


図59

【図60】

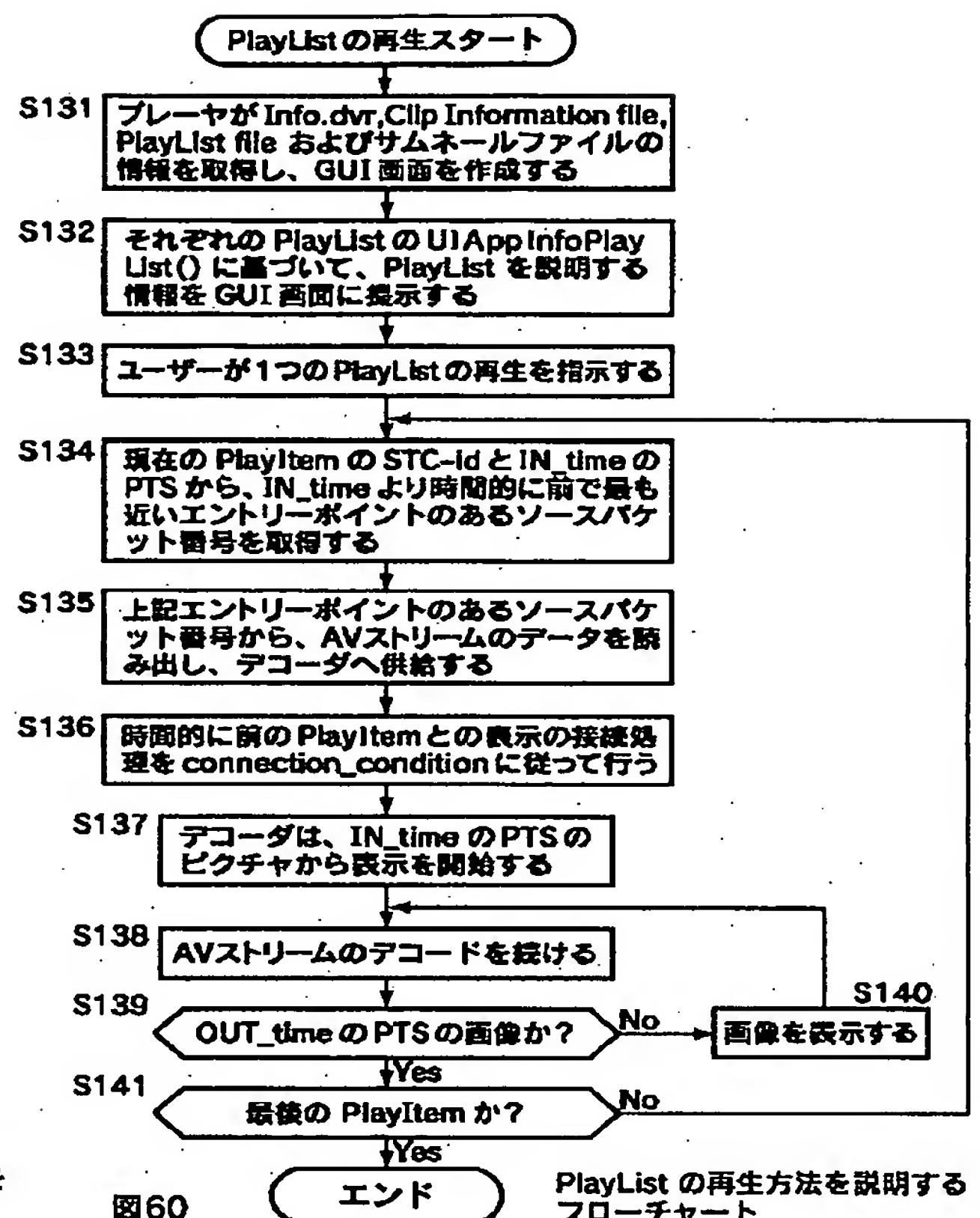
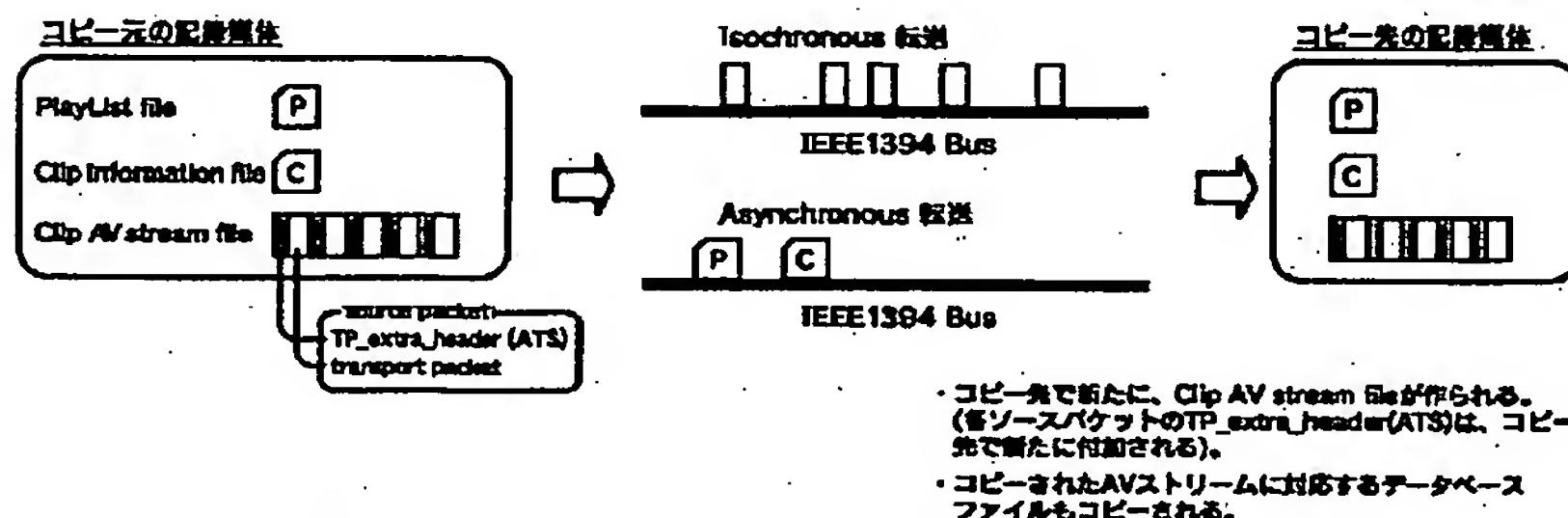


図60

【図62】

AVストリームをリアルタイム転送(ストリーム転送)し、データベースをファイル転送する場合

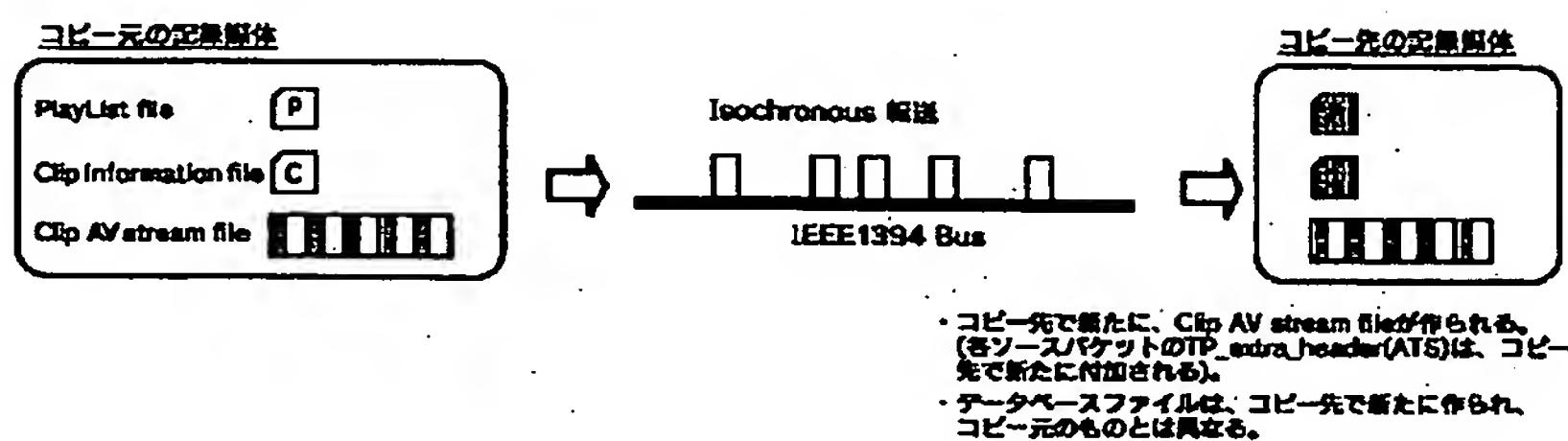


AVストリームをリアルタイム転送(ストリーム転送)し、データベースをファイル転送する場合を説明する図

図62

【図63】

AVストリームをリアルタイム転送(ストリーム転送)する場合

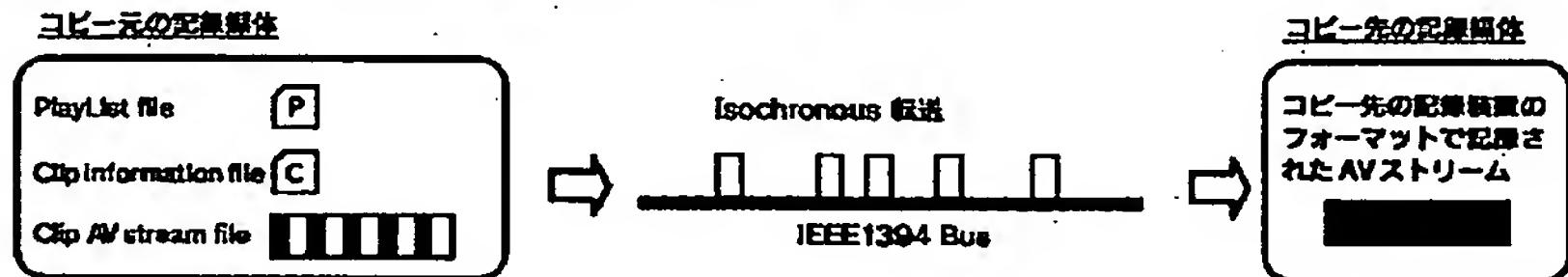


AVストリームだけをリアルタイム転送し、コピー先でデータベースファイルを新たに作成する場合を説明する図

図63

【図64】

AVストリームをリアルタイム転送(ストリーム転送)する場合

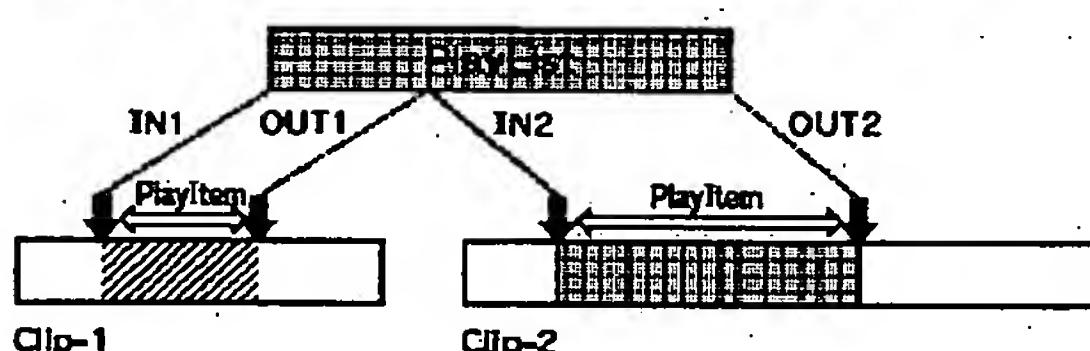


コピー先の記録装置がDVRフォーマットに準拠したものでない場合に、AVストリームをリアルタイムにデータを再生する速度で転送する場合を説明する図

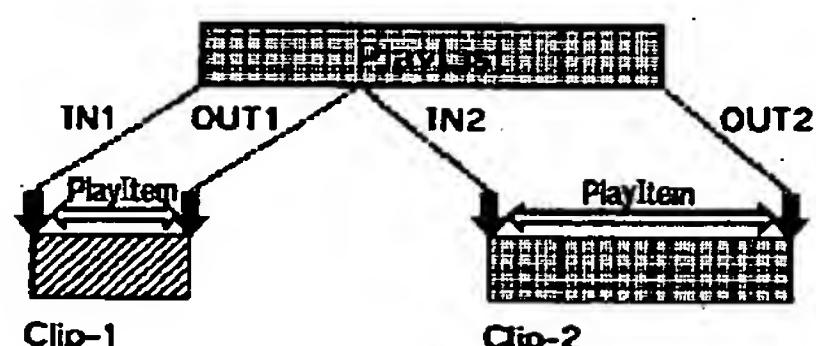
図64

【図65】

コピー元(出力側の再生装置)で記録されているファイル



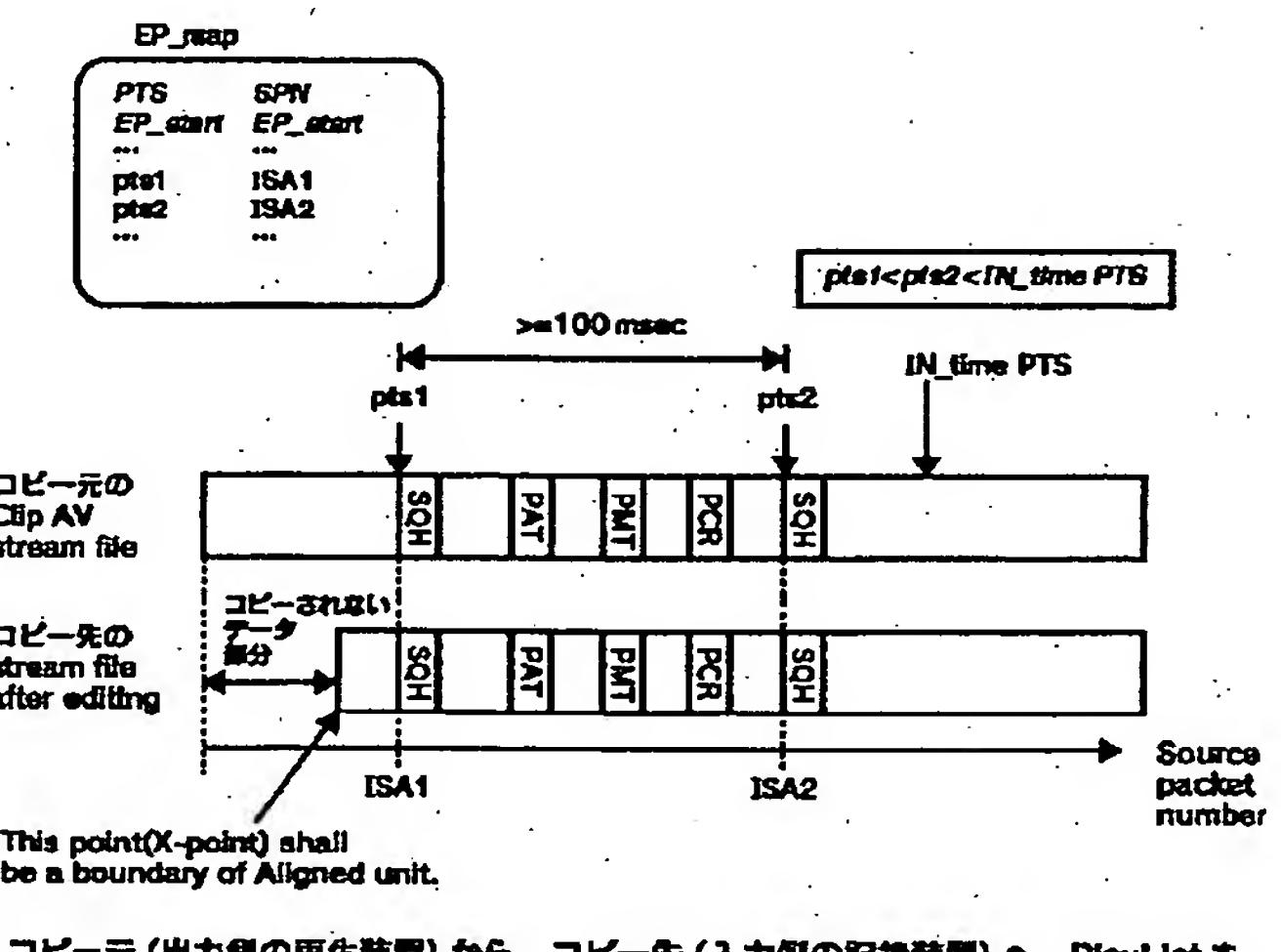
コピー先(入力側の記録装置)で記録されるファイル



コピー元(出力側の再生装置)から、コピー先(入力側の記録装置)へ、PlayListとそれに必要なClipの部分だけをコピーする場合のClipとPlayListの関係を説明する図

図65

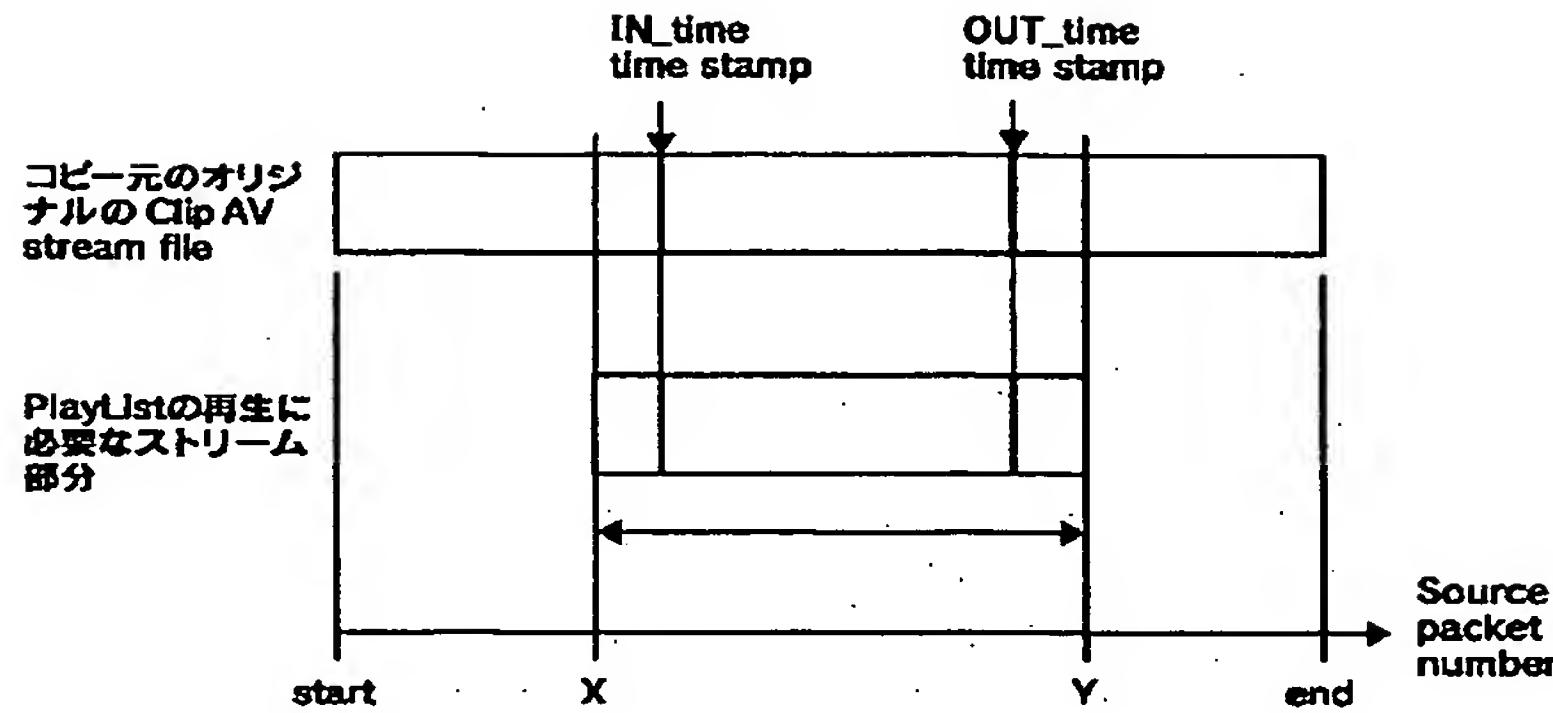
【図67】



コピー元(出力側の再生装置)から、コピー先(入力側の記録装置)へ、PlayListをコピーする場合に、IN_timeの前のデータのコピー開始点を決める方法を説明

図67

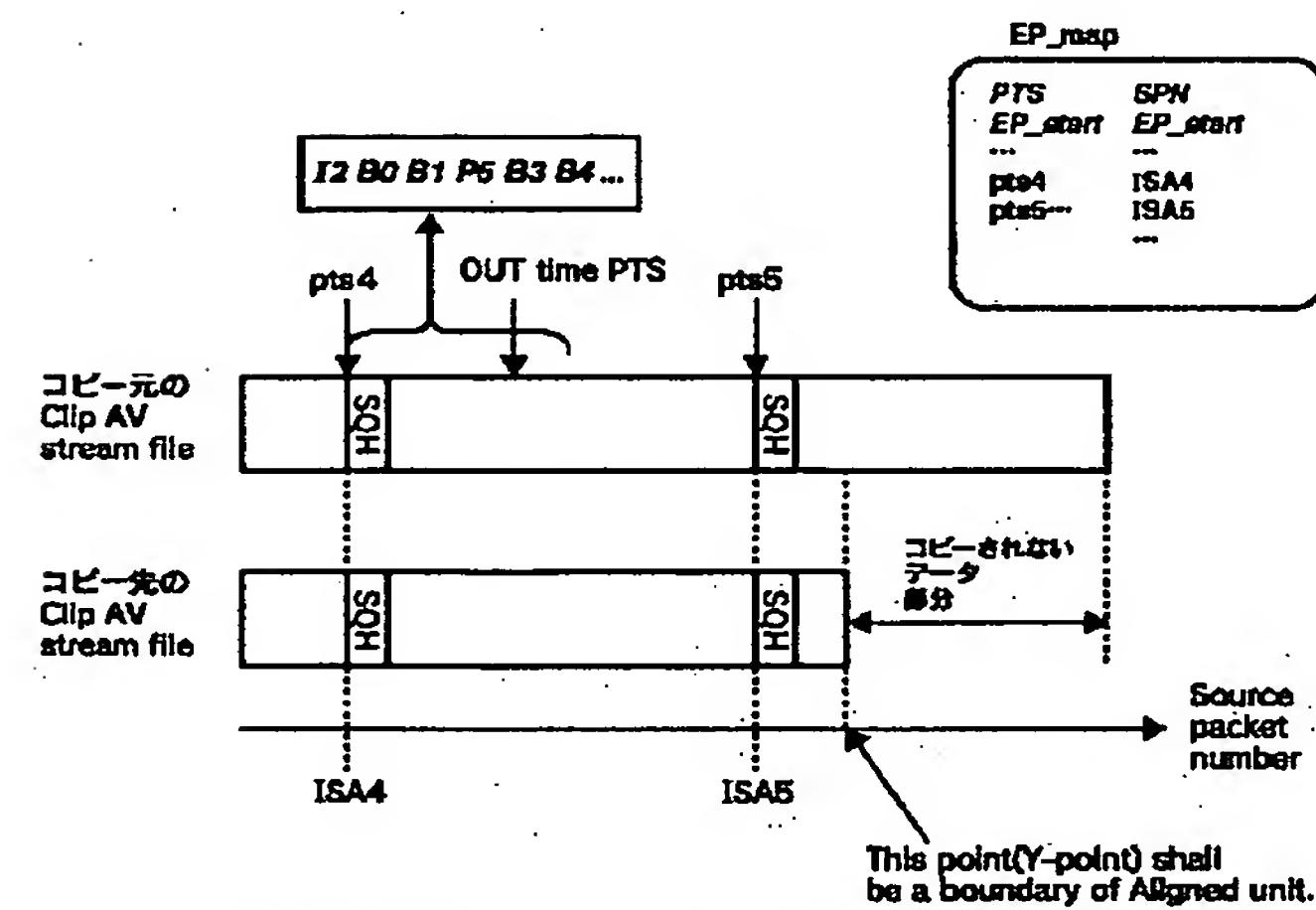
【図66】



コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、
Playlistをコピーする場合に、そのPlaylistの再生に必要なClipの部分について
説明する図

図66

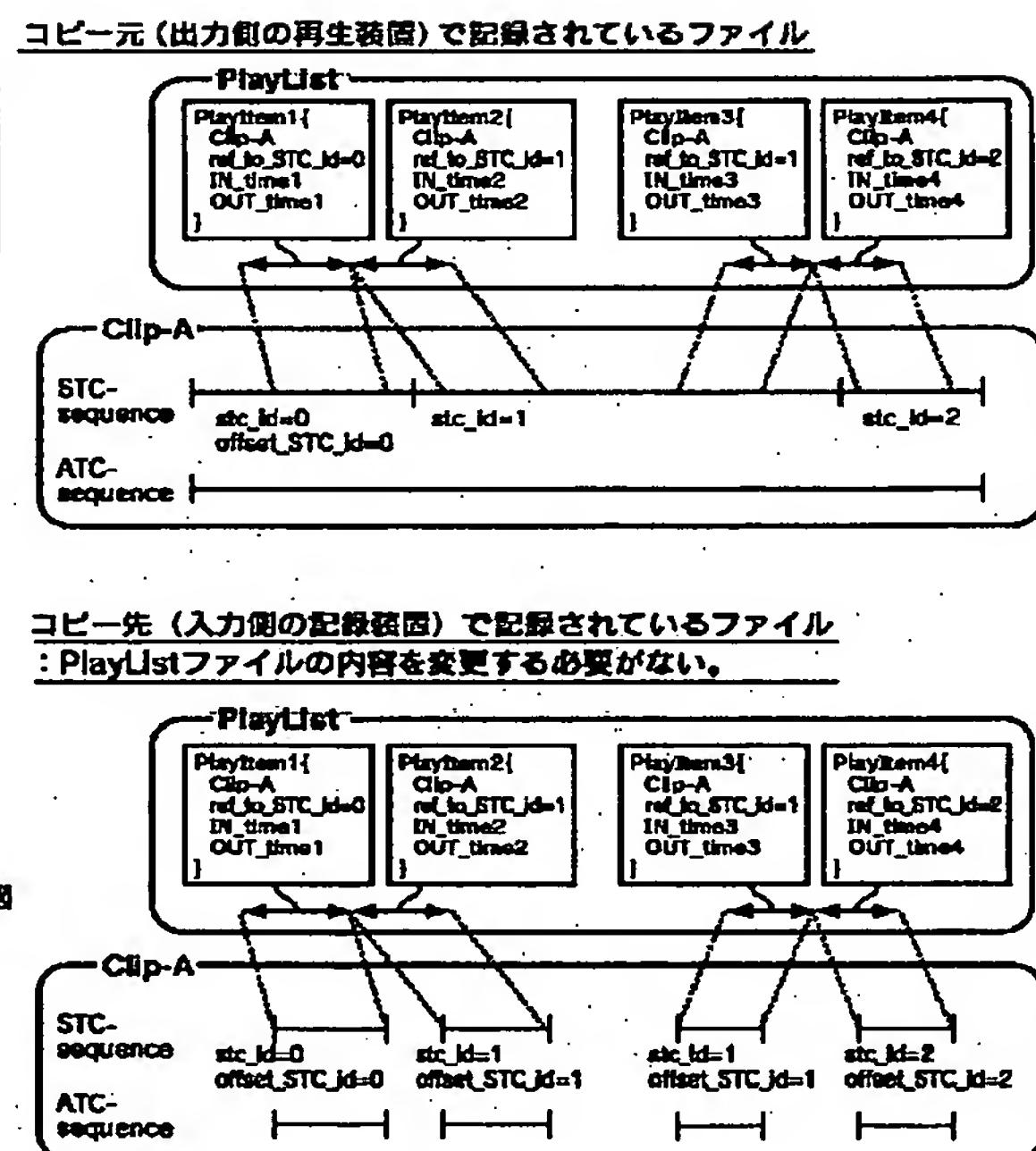
【図68】



コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、Playlistを
コピーする場合に、OUT_time の後のデータのコピー終了点を決める方法を説明する図

図68

【図69】

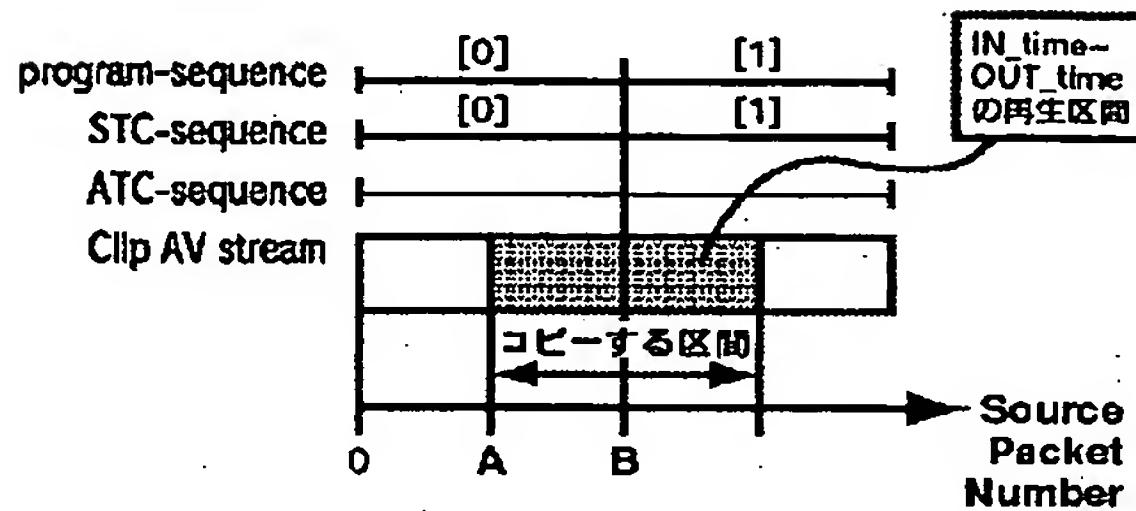


コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、Playlist
とそれに必要なClipの部分だけをコピーする場合の例であり、その時のClipと
Playlistの関係を説明する図

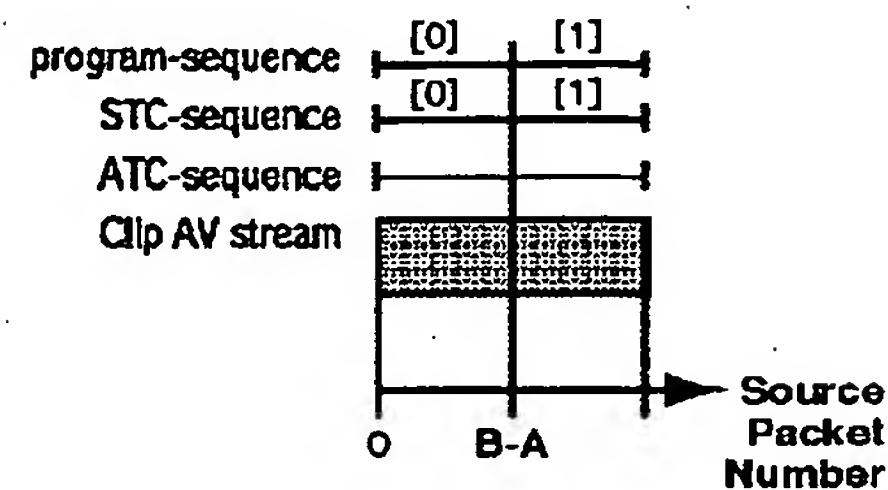
図69

【図70】

(A) コピー元で記録されている Clip



(B) コピー先へ転送する Clip



(C) コピー元 (出力側の再生装置) で記録されている Clip Informationファイル

Clip Information file

- $SPN_ATC_start[0] = 0$
- $SPN_STC_start[0] = 0, SPN_STC_start[1] = B$
- $SPN_program_sequence_start[0] = 0, SPN_program_sequence_start[1] = B$
- EP_map
- ClipMark

| PTS_EP_start | SPN_EP_start |
|--------------|--------------|
| pts(x1) | X1 |
| pts(x2) | X2 |
| ... | ... |
| pts(xa) | Xa |
| pts(xb) | Xb |
| ... | ... |
| pts(xz) | Xz |
| ... | ... |

コピーするストリーム
区間が使用するデータ
 $IN_time < pts(xa),$
 $pts(xz) < OUT_time$

| mark_type | mark_time_stamp |
|-----------|-----------------|
| type_A | pts(A) |
| type_B | pts(B) |
| ... | ... |
| type_F | pts(F) |
| type_S | pts(S) |
| ... | ... |
| type_Q | pts(Q) |
| ... | ... |

コピーするストリーム
区間が使用するデータ
 $IN_time < pts(F),$
 $pts(Q) < OUT_time$

(D) コピー先 (入力側の記録装置) へ転送する Clip Informationファイル

Clip Information file

- $SPN_ATC_start[0] = 0$
- $SPN_STC_start[0] = 0, SPN_STC_start[1] = B-A$
- $SPN_program_sequence_start[0] = 0, SPN_program_sequence_start[1] = B-A$

• EP_map

| PTS_EP_start | SPN_EP_start |
|--------------|--------------|
| pts(xa) | Xa-A |
| pts(xb) | Xb-A |
| ... | ... |
| pts(xz) | Xz-A |

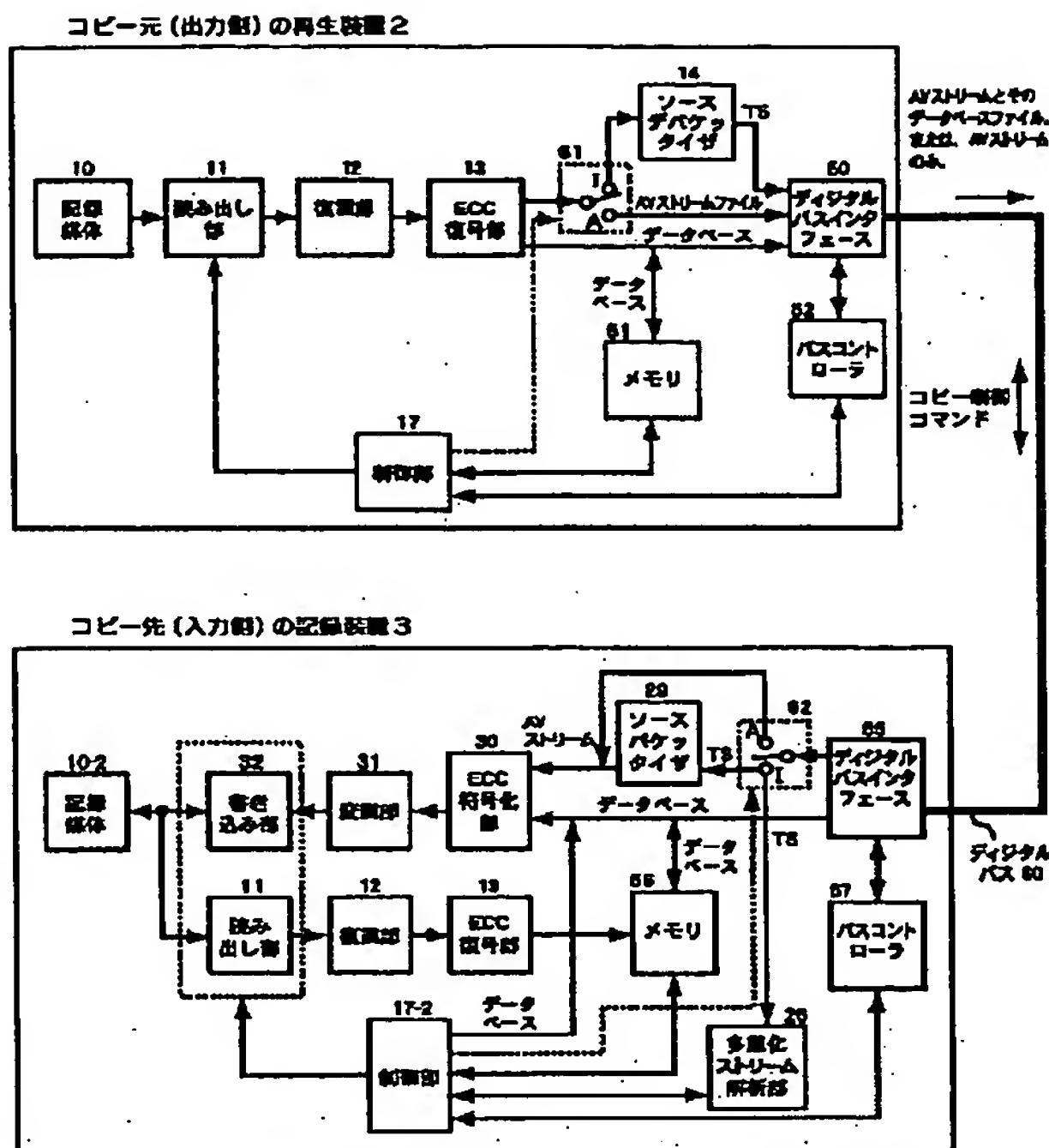
• ClipMark

| mark_type | mark_time_stamp |
|-----------|-----------------|
| type_F | pts(F) |
| type_S | pts(S) |
| ... | ... |
| type_Q | pts(Q) |

コピー元 (出力側の再生装置) から、コピー先 (入力側の記録装置) へ、
Clip を部分的にコピーする場合の Clip の変更方法について説明する図

図70

【図71】



コピー元（出力側の再生装置）から、コピー先（入力側の記録装置）へ、DVRのAVストリームファイルとそれに同様するデータベースファイルを、デジタルバス経由でコピーする場合の構成を示すブロック図

図71

【図72】

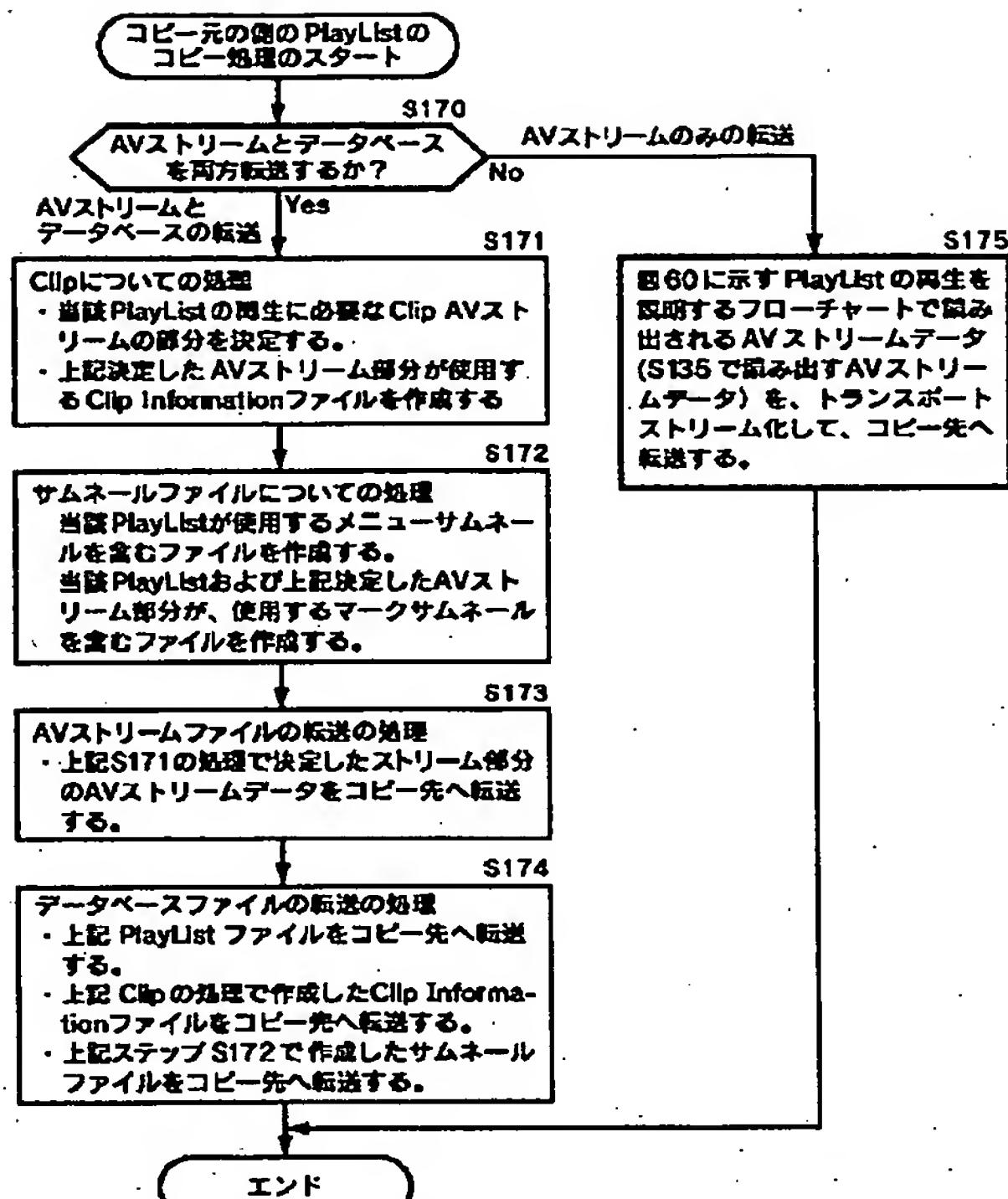
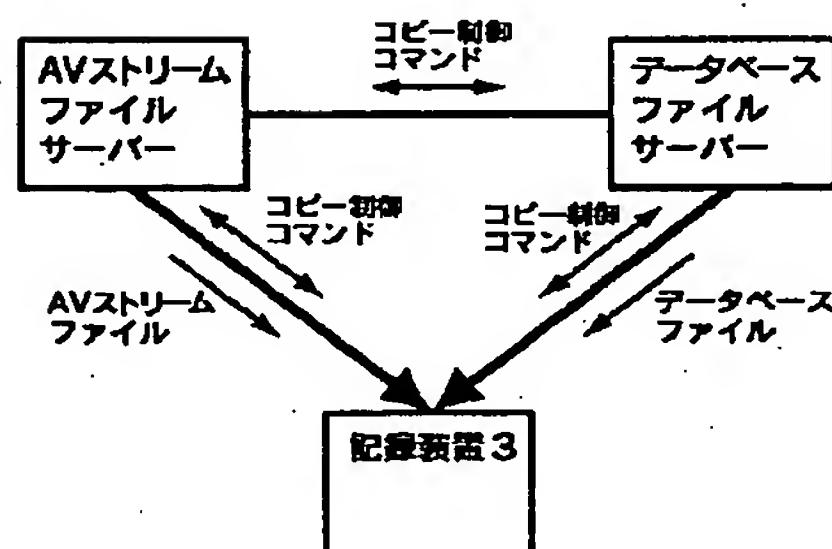


図72

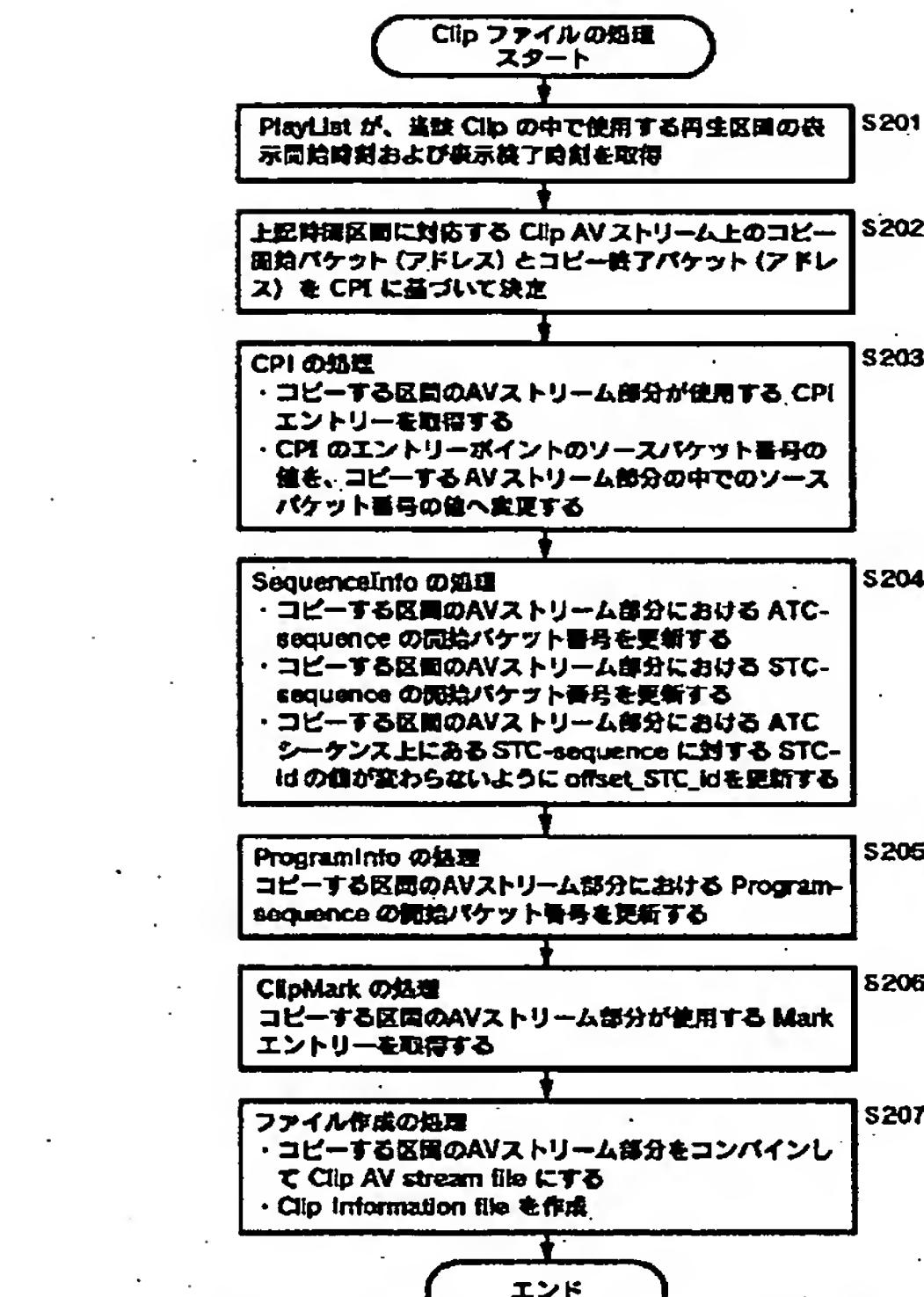
【図76】



AVストリームファイルとデータベースファイルが、別々のサーバーに記録されており、それぞれのサーバーから記録装置へファイルが転送される場合

図76

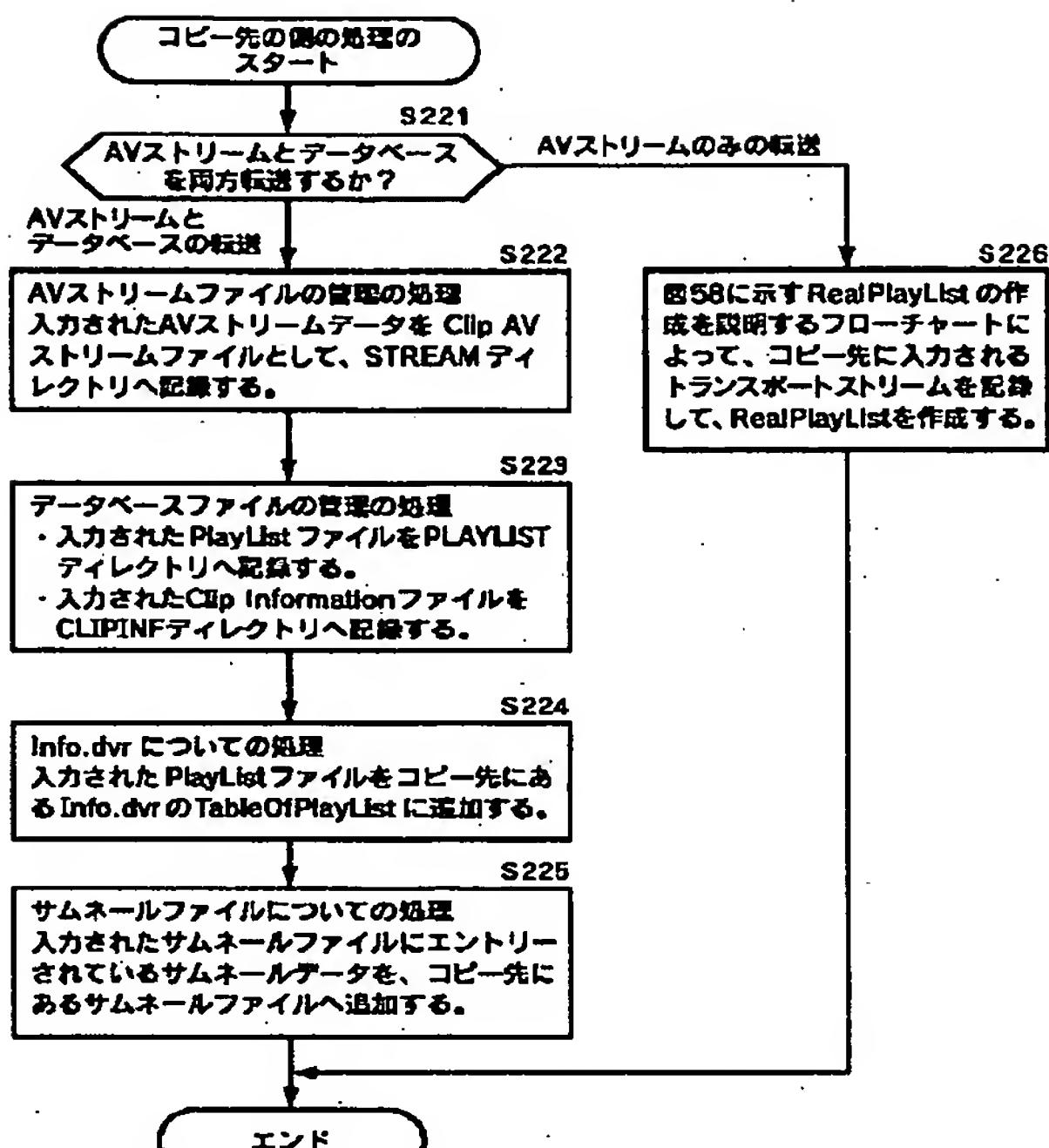
【図73】



コピー元(出力側の再生装置)から、コピー先(入力側の記録装置)へ、PlayListをコピーする場合において、コピー元の側のClipについての処理を説明するフローチャート

図73

【図74】



コピー元(出力側の再生装置)から、コピー先(入力側の記録装置)へ、PlayListをコピーする場合の、コピー先の処理を説明するフローチャート

図74

フロントページの続き

(51) Int. Cl. 7

H 0 4 N 5/765
5/85
5/91
5/92

識別記号

F I
H 0 4 N 5/85
5/91
5/92
5/91

テーマコード(参考)
Z 5 K 0 2 8
P
H
L

F ターム(参考) 5C052 AA02 AB03 AB05 AC08 CC06
CC11 DD04
5C053 FA06 FA15 FA23 GA11 GB06
GB38 HA29 HA31 JA22 JA24
KA07 KA18 KA20 LA15
5D044 AB05 AB07 BC03 CC06 DE24
DE38 DE39 DE53 FG19 GK08
GK12 HL11
5D077 AA23 BA14 BA18 CA02 DC03
DC08 EA34
5D110 AA15 AA27 AA29 DA03 DA11
DA15 DA17 DA20 DB03 DC05
DE01
5K028 AA12 AA14 EE03 KK32 MM12
NN01